T E S Report Number: C204-1

8 - 9018

CONTRACT SERIAL NO. OEZ78-05111
CONTRACTOR: TES Limited
REQUISITION NO. 23440-8-9018

# PILOT HYDROCRACKING ONLINE INFORMATION LOGGING (PHOIL) SYSTEM SOFTWARE

May 1979

Prepared For

Department of Energy, Mines and Resources
CANMET

By

T E S Limited

Ottawa, Canada

109 pp

MICROMEDIA

TABLE OF CONTENTS


| | Page No. |
|---|---|
| 1. References | 1 |
| 2. Introduction | 2 |
| 3. System Description | 3 |
| 3.1 General Description | 3 |
| 3.2 Allocation of Channels | 5 |
| 3.3 System Tape Layout | 5 |
| 3.4 System Variables, Tables, Constants and Data Files | 8 |
| 3.5 Data Tape Layout | 17 |
| 3.6 Allocation of Device Select Codes | 19 |
| 3.7 System Flow Charts | 20 |
| 4. Program Flow Charts and Source Listings | 24 |
| 4.1 Introduction | 24 |
| 4.2 File 0 - Initialization and Power on Start | 25 |
| 4.3 File 3 - Scanning and Data Storage | 32 |
| 4.4 File 4 - Operation Changes | 46 |
| 4.5 File 5 - Edit Options | 64 |
| 4.6 File 6 - Program Fluke Channel Functions | 77 |
| 4.7 File 10 - Program for Copying Data Files | 81 |
| 5. Considerations for Interfacing HP 9830A with 11205A Interface to Fluke 2240B Data Logger with -17 Option | 83 |
| 6. Data Extraction and Plotting Programs | 86 |
| 6.1 Introduction | 86 |
| 6.2 System Flow Charts | 87 |
| 6.3 Data Extraction Program | 88 |
| 6.4 Plotter Program | 100 |

## 1. References

The following documents supply additional information pertinent to a proper understanding of the contents of this volume.

1. PHOIL Operations Manual
2. Hewlett-Packard 9830A Calculator Operating & Programming Manual
3. Hewlett-Packard 9830A Calculator 11274B String Variables ROM Operating Manual
4. Hewlett-Packard 9830A Calculator 11274B Extended I/O ROM Operating Manual
5. Hewlett-Packard 9830A Calculator Plotter Control ROM Operating Manual for 1127B and Option 271
6. Fluke Instruction Manual Model 2240B

## 2. Introduction

This report describes calculator software developed by T E S Limited for the Department of Energy, Mines and Resources, CANMET, to acquire data during operation of the two pilot hydrocracking plants situated at the CANMET Bells Corners, Ontario facility. The main hardware components had been previously acquired by CANMET and the major ones are a Hewlett-Packard HP9830A desktop programmable calculator and a Fluke 2240B data logger. During the course of the work reported on, several difficulties were encountered with the hardware. These were traced to the Fluke data logger and after some delay this instrument was put into serviceable condition by the manufacturer. Some peculiarities remain in the way in which the Fluke data logger and the HP calculator must be interfaced to each other. These peculiarities involve some pin changes in interconnecting wiring and a few awkward programming protocols which must be followed. These peculiarities are discussed in Chapter 5.

Chapter 3 provides an overview of the system and describes the contents and layout of the system operation tape and the format of the data tapes. Chapter 4 contains descriptions, flow charts and source listings of each program. Chapter 5 contains notes on interfacing between the datalogger and calculator. Chapter 6 contains notes, flow charts and source listings for separate programs which retrieve data from the data tapes and plot it. These latter programs are provided on a separate tape.

## 3. System Description

### 3.1 General Description

During operation of the hydrocracking plants, a number of sensors (thermocouples, pressure transducers, etc.) continuously monitor the state of the plants, rates of flow and other variables. In order to analyse the results of experiments on the processes used, it is desirable to record the outputs of these sensors for later study. To this end hardware was acquired consisting of a Fluke data logger which scans each sensor in turn and provides a digital output of its reading, and a Hewlett-Packard desktop programmable calculator which is interfaced with the data logger and performs a number of functions.

The primary function of the calculator is to interrogate the data logger to obtain the data from each successive complete scan and to store this data periodically on an external cassette tape in a recoverable format. In addition, the calculator scales the output of some channels in order that the system operator can monitor the output, if desired, in meaningful units. The calculator also performs some calculations of parameters which are not directly measurable by the sensors and can check each channel against a table of limits and print out an alarm message if these limits are exceeded. In order to reduce false alarms, out-of-limit conditions are verified before the alarm is printed. There is provision in the program for the operator to enter, at the keyboard, certain data which cannot readily be obtained through the use of automatic sensors, including comments, if desired.

Throughout these operations the calculator operates in a controller mode, initiating scans by the data logger and controlling the channels to be scanned, etc. In the event of a power failure, the system will

recover automatically when the power comes back on. This is accomplished by means of a hardware option on the calculator which automatically loads and runs File 0 on the system tape. This file contains a program which initializes all necessary tables and flags correctly and then passes control to the regular operating program. The mechanism also provides a good means of recovery from hardware malfunctions or a serious operator error. All that is required is that the calculator main power switch be turned off momentarily. This automatic program also starts the system when it is initially turned on.

The programs are modular in nature and system parameters which are subject to changes (such as, for example, channel allocations for the data logger, or channel alarm limits) are contained in tables which are easy to edit without interfering with the operating programs. The system runs automatically except when an operator wishes to interact with it in order to enter manually taken data, change alarm limits, have the readings from one or more channels printed out continuously, change scaling factors or units or indicate a change in the operating mode of a plant. In these cases, an interrupt switch, which is scanned by the data logger, is flipped and on the next scan the calculator goes into an interactive mode which leads the operator through the steps necessary to make the required changes or enter data.

The calculator stores the scaled scan data for the scan which immediately follows a change in the hour and includes the calculated values as well. Data which is manually input by the operator is stored with the appropriate hourly scan data as indicated by the operator. The data is stored on external cassette tapes and the data pertaining to the operation of each of the pilot plants is stored separately. The operator may set a system flag so that every scan is stored instead of hourly scans. Also, the first scan after a plant start-up is stored as is the last scan before a plant shut-down.

## 3.2 Allocation of Channels

The Fluke data logger can scan up to 180 channels, numbered 0 to 179. Channel 0 is used for the interrupt switch which allows the operator to change the system operation as required. The channels are divided so that Channels 1 to 89 pertain to pilot plant number 1 and Channels 90 to 179 pertain to pilot plant number 2. Channels 1 to 59 and 120 to 179 are designated as direct temperature readings and there is no provision for scaling them. Channels 60 to 119 may be scaled according to the formula

$$Ax + B$$

where A and B are scale factors which may be unique for each channel. Each of Channels 60 to 119 are scaled before storage.

In the calculator programs, an additional 21 channels are provided to hold the results of calculations of values from the scan values. Channels 180 to 189 are allotted to pilot plant 2 and Channels 190 to 200 are allotted to pilot plant 1. This results in a total of 100 channels for each plant. The specific allocations are shown in Section 3.4 below. Some channels are not allocated to allow for future changes.

## 3.3 System Tape Layout

The system tape contains 9 files numbered from 0 to 7 plus 10.

File 0 is a program file which contains a program to initialize the system when the calculator is turned on or after a power failure. It initializes constants and tables, reads Files 1 and 2 (which contain current operating flags, parameters and limits) into memory, initializes all variables and then loads and runs File 3 (which is the main operating program). The constants are loaded into tables in this way to facilitate future changes in calculations, etc.

File 1 is a data file containing the variable C(200,3) which contains the centre value and ± limits for the alarm points for each channel, the conversion factors A and B for each of Channels 60 to 119 (see Section 3.2 above) and some system flags and other data. The specific contents of this file are given in Section 3.4 below. This file is loaded into memory by File 0 at the "power-on start" point. Programs in Files 3, 4 and 5 all make changes to this data at some point and then store the updated file.

File 2 is a data file containing the variable E(200,3) which contains the unit codes for each channel, the print flags for each channel (indicating whether the channel in question should be printed on each scan or not) and the directory of stored data files which indicates their status. This file is loaded into memory by File 0 at the "power-on start" point. Programs in Files 3, 4 and 5 all make changes to this data at some point and then store the updated file.

File 3 is a program file which contains the main scanning program. It is linked by File 0 after initialization is complete. It links Files 4 and/or 5 during the interrupt service routine. File 4 links File 3 at the "Return to Scan" point. File 5 links File 3 at the "Update File and Exit" point. File 3 stores data in File 1, C(200,3) after calculating the feed rate (2 points) and stores data in File 1, C(200,3) and File 2, E(200,3) after calculating ~~the recycle gas rate~~ and on an intentional power-down. File 3 stores the data from a scan after the hour changes or on every scan (if F8 = 0) at "Hourly Data Storage" point.

Data from plant 1 is stored on the cassette tape drive designated by Device Select Code 5. Data from plant 2 is stored on the cassette tape drive designated by Device Select Code 6.

File 4 is a program file containing several of the programs which are run under interrupt. This file contains the programs for plant start-up,

plant shut-down, cassette tape changing, plant status change, product barrel change, manual data input and hourly report. File 4 stores data in File 1, C(200,3) and File 2,E(200,3) at the "Return to Scan" point where it also links File 3. It is initially brought into memory by File 3 operating under interrupt.

File 5 is a program file containing the programs which are run under interrupt and allow the operator to change conversion factors, alarm limits or units, or to monitor one or more channels continuously. File 5 stores data in File 1, C(200,3) and File 2, E(200,3) at the "Update File and Exit" point where it also links File 3. File 5 is linked by File 3.

File 6 is a program file which allows programming of the channel functions on the Fluke data logger. It operates independently of the rest of the system.

File 7 is a data file which is used by File 6 to store a version of the Fluke data logger program. The program in File 6 can also compare the current program in the Fluke data logger with the version in File 7 and note discrepancies.

File 10 contains a utility program which eases the task of copying data files and is useful, therefore, in making back-up copies of the system tape.

## 3.4 System Variables, Tables, Constants and Data Files

| Variables | Use | Details |
|---|---|---|
| A(200) | internal array of raw data, converted data and calculated values | Section 3.2 & Table 1 |
| C(200,3) | array of alarm limits, conversion factors and system flags and data. up-to-date copy maintained at all times in system tape File 1 | Section 3.3 & Table 2 |
| E(200,3) | array of unit codes, array of print flags and directory of data file status | Section 3.3 & Table 3 |
| P(130) | array of final data as stored on the data tapes for plants 1 and 2. contents vary slightly for each plant | Tables 4 & 5 |
| N$(20) | Run number | Section 3.5 |
| T$(80) | Comments | Section 3.5 |
| U$(80) | Comments | Section 3.5 |
| V$(80) | Comments | Section 3.5 |
| H$(20) | Time | Section 3.5 |

TABLES

W(3,3):

This table directs the plant status routine to the proper routine depending on the old status and the new status. The status codes are as follows: 0 - Plant running an experiment with alarm function on.

1 - Plant between runs with data storage but with no alarm checking

2 - Plant shut-down, no data storage

| Status | | Entry | Subroutine |
|---|---|---|---|
| Old | New | | |
| 0 | 0 | 3 | Just update |
| 0 | 1 | 1 | Shut-down |
| 0 | 2 | 1 | Shut-down |
| 1 | 0 | 2 | Start-up |
| 1 | 1 | 3 | Just update |
| 1 | 2 | 3 | Just update |
| 2 | 0 | 2 | Start-up |
| 2 | 1 | 3 | Just update |
| 2 | 2 | 3 | Just update |

L(6,6):

This table holds the channel numbers of the variables in the heat transfer coefficient and heat transfer rate equations. The first subscript gives the position of the variable in the equation. The second subscript identifies the physical location on one of the reactors for which the calculation is being performed.

T(4,2):

This table holds the channel numbers where variables used in the feed rate calculations can be found. The second subscript refers to the plant number, the first is the name of the variable and is clearly labeled in the program in File 0.

G(6,2):

This table holds the channel numbers where variables used in the recycle gas rate calculation can be found. The second subscript is the plant number and the first is the variable name.

CONSTANTS:

K1 - K5:

These are constants used in various equations throughout the program.

FLAGS:

F1 Print time and date
F2 All print flags clear
F5 Interrupt
F6 Not input wt
F8 Not store every scan
F9 Not power-on start

TABLE 1

| SCAN DATA | Variable: DIM A(200) |
|---|---|
| A(1) - A(179) | CHANNEL DATA (60 - 119 converted) |
| A(180) | HTC 5 Plant 2 |
| A(181) | HTC 4 Plant 2 |
| A(182) | HTC 2 Plant 2 |
| A(183) | HTR 5 Plant 2 |
| A(184) | HTR 4 Plant 2 |
| A(185) | HTR 2 Plant 2 |
| A(186) | Not allocated Plant 2 |
| A(187) | Recycle Gas Rate Plant 2 |
| A(188) | Feed Rate Plant 2 |
| A(189) | Not allocated |
| A(190) | HTC 5 Plant 1 |
| A(191) | HTC 4 Plant 1 |
| A(192) | HTC 2 Plant 1 |
| A(193) | HTR 5 Plant 1 |
| A(194) | HTR 4 Plant 1 |
| A(195) | HTR 2 Plant 1 |
| A(196) | Not allocated Plant 1 |
| A(197) | Recycle Gas Rate Plant 1 |
| A(198) | Feed Rate Plant 1 |
| A(199) | Not allocated |
| A(200) | Not allocated |

TABLE 2

Variable: DIM C(200,3)

| | |
|---|---|
| C(Channel,1) | Alarm point centre value |
| C(Channel,2) | Alarm point ± value |
| C(1,3) to C(60,3) | Conversion factors A (Channels 60 to 119, see Section 3.2) |
| C(61,3) to C(120,3) | Conversion factors B " |
| Plant number 1 data and flags. | |
| C(121,3) | Initial feed wt. 1 |
| C(122,3) | " 2 |
| C(123,3) | N (Number of removed weights) |
| C(124,3) | Old SW |
| C(125,3) | Old wt. |
| C(126,3) | Feed system total |
| C(127,3) | Current feed |
| C(128,3) | LE tare |
| C(129,3) | HE tare |
| C(130,3) | LE sample |
| C(131,3) | HE sample |
| C(132,3) | LE system total |
| C(133,3) | HE system total |
| C(134,3) | LE old barrel final wt. |
| C(135,3) | HE old barrel final wt. |
| C(136,3) to C(142,3) | Not allocated |
| C(143,3) to C(150,3) | Array of alternating data (accumulated feeds) and times |

TABLE 2 (cont'd)

Plant number 2 data and flags.

| | |
|---|---|
| C(151,3) | Initial feed wt.1 |
| C(152,3) | Initial feed wt.2 |
| C(153,3) | N = 0 (Constant) |
| C(154,3) | Old SW |
| C(155,3) | Old wt |
| C(156,3) | Feed system total |
| C(157,3) | Current feed |
| C(158,3) | LE tare |
| C(159,3) | HE tare |
| C(160,3) | LE Sample |
| C(161,3) | HE sample |
| C(162,3) | LE system total |
| C(163,3) | HE system total |
| C(164,3) | LE old barrel final wt. |
| C(165,3) | HE old barrel final wt. |
| C(166,3) to C(172,3) | Not allocated |
| C(173,3) to C(180,3) | Array of alternating data (accumulated feeds) and times |

Both plants

| | |
|---|---|
| C(181,3) | Year (YYYY) |
| C(182,3) | Date (Day of year, 1 to 366) |
| C(183,3) | Time (Seconds of the day) |
| C(184,3) | Status of plant 1 |
| C(185,3) | Status of plant 2 |
| C(186,3) | Hour of last scan |
| C(187,3) | Last day of the year |
| C(188,3) to C(200,3) | Not allocated |

TABLE 3

Variable: DIM E(200,3)

| | |
|---|---|
| E(Channel, 1) | Unit codes for each channel |
| E(Channel,2) | Print flags for each channel (if flag = 1 then the channel value will be printed on each scan) |
| E(File,3) | Output data file status<br>The value for each file is in the format HHDDD where HH is the hour of the day and DDD is the day of the year. If value = 0, the file is empty. When scan data is written to the file, the value is updated to the hour and day of writing. When the manually-entered data, corresponding to the scan data, is entered, 366 is added to the day of the year to indicate that the file is complete. The special value 25400 is used to identify the start-up file and the special value 26400 is used to identify the shut-down file. |

TABLE 4

| FINAL DATA, Plant 1 | Variable: COM P (130) |
|---|---|
| P(1) to P(89) | Channels 1 - 89, A(1) to A(89) |
| P(90) to P(100) | Channels 190 - 200, A(190) to A(200) |
| P(101) | Initial feed wt.1 |
| P(102) | Initial feed wt. 2 |
| P(103) | N |
| P(104) | Old SW |
| P(105) | Old wt |
| P(106) | Feed system total |
| P(107) | Current feed |
| P(108) | LE tare |
| P(109) | HE tare |
| P(110) | LE sample |
| P(111) | HE sample |
| P(112) | LE system total |
| P(113) | HE system total |
| P(114) | LE Final wt. of old barrel |
| P(115) | HE Final wt. of old barrel |
| P(116) | Not allocated |
| P(117) | ID No. |
| P(118) | LE Off gas volume |
| P(119) | HE Off gas volume |
| P(120) | Scrubber Off gas volume |
| P(121) | Recycle Off gas volume |
| P(122) | Reactor Watts |
| P(123) | Preheater Watts |
| P(124) | LE Present wt. |
| P(125) | HE Present wt. |
| P(126) | Year (YYYY) |
| P(127) | Date (Day of year 1 -366) |
| P(128) | Time of day (sec) |
| P(129) | Status P1 |
| P(130) | Status P2 |

TABLE 5

| FINAL DATA, Plant 2 | Variable: COM P(130) |
|---|---|
| P(1) to P(90) | Channels 90 - 179, A(90) to A(179) |
| P(90) to P(100) | Channels 180 - 189, A(180) to A(189) |
| P(101) | Initial feed wt. 1 |
| P(102) | Initial feed wt. 2 |
| P(103) | N = 0 |
| P(104) | Old SW |
| P(105) | Old wt |
| P(106) | Feed system total |
| P(107) | Current feed |
| P(108) | LE tare |
| P(109) | HE tare |
| P(110) | LE sample |
| P(111) | HE sample |
| P(112) | LE system total |
| P(113) | HE system total |
| P(114) | LE Final wt. of old barrel |
| P(115) | HE Final wt. of old barrel |
| P(116) | Not allocated |
| P(117) | ID No. |
| P(118) | LE Off gas volume |
| P(119) | HE Off gas volume |
| P(120) | Scrubber Off gas volume |
| P(121) | Recycle Off gas volume |
| P(122) | Reactor Watts |
| P(123) | Preheater Watts |
| P(124) | X Watts |
| P(125) | Y Watts |
| P(126) | Year (YYYY) |
| P(127) | Date |
| P(128) | Time |
| P(129) | Status P1 |
| P(130) | Status P2 |

## 3.5 Data Tape Layout

A data tape contains hourly scan and manually input data for the operation of one of the two pilot plants. There is room on a data tape for approximately 50 files. A file of data (start-up file) is stored when a plant is switched from mode 1 (alarms off but data being stored) or mode 2 (shut-down) to mode 0 (full operation). A file of data (shut-down file) is also stored when a plant is switched out of mode 0. Thus there is provision for recording data during periods between experiments (mode 1) while plant operation is changing as well as for clearly defining initial and final conditions of an experiment and recording data hourly throughout the experiment. For periods of critical operation, it is possible to switch to a mode which stores data from every scan. Note that this mode will completely fill a fresh data tape in a relatively short time.

All of the above changes in plant mode or storage mode are accomplished by using the interrupt switch and then choosing the appropriate menu item when prompted by the calculator.

The operator is prompted to "Change Cassette" when 50 files have been recorded on a data tape. The program will, however, continue to record data until the tape runs out even if the tape is not changed immediately. The "Change Cassette" warning will be printed each time data is stored until the tape is, in fact, changed.

Each data file consists, in order, of the variables P(130), H$(20), N$(20), T$(80), U$(80), V$(80). P(130) is described in detail in Section 3.4 above. H$(20) contains an alphanumeric representation of the time. N$(20) contains an alphanumeric representation of the run identifier used to label the experiment. T$, U$ and V$ allow the operator to include up to 240 alphanumeric characters of comments with each hour's data. These comments are entered at the same time as the manually input data.

When a cassette is changed, a copy of the directory (variable E(*,3)) for the data tape is placed in File 0 of the tape.

Note that failure to change a data tape before it runs out completely will probably result in a system hang-up with a resulting loss of data and monitoring.

## 3.6 Allocation of Device Select Codes

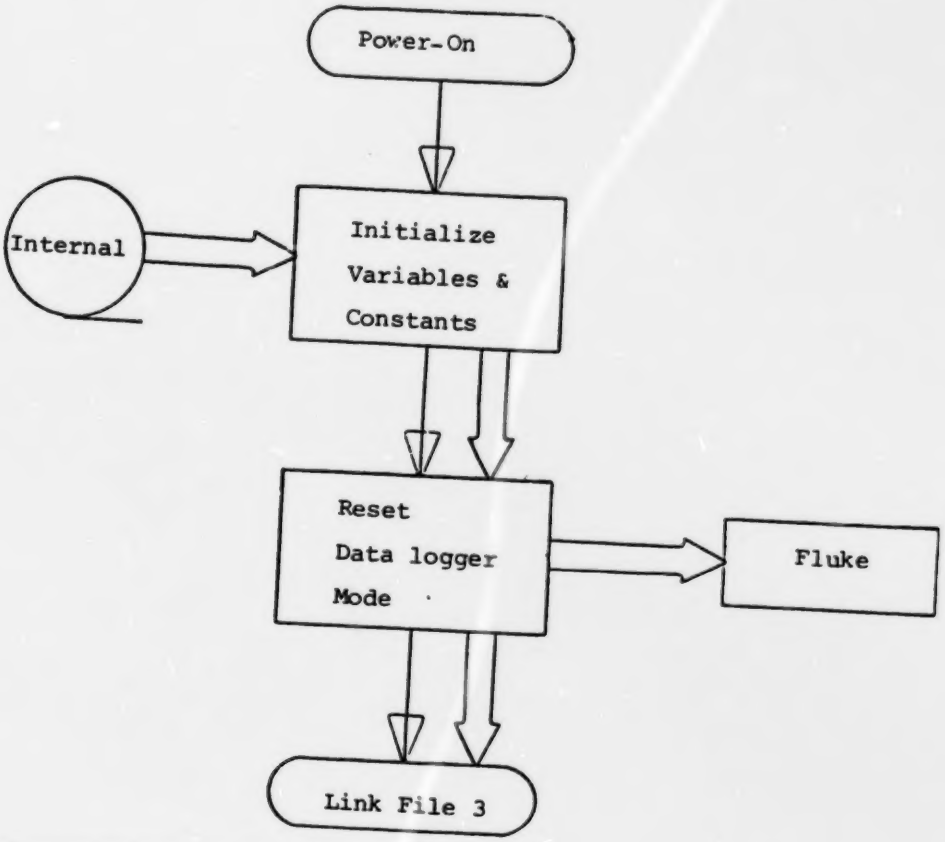
Data from plant 1 is stored on the cassette tape drive designated by Device Select Code 5. Data from plant 2 is stored on the cassette tape drive designated by Device Select Code 6.

The Fluke data logger is designated by Device Select Code 9.

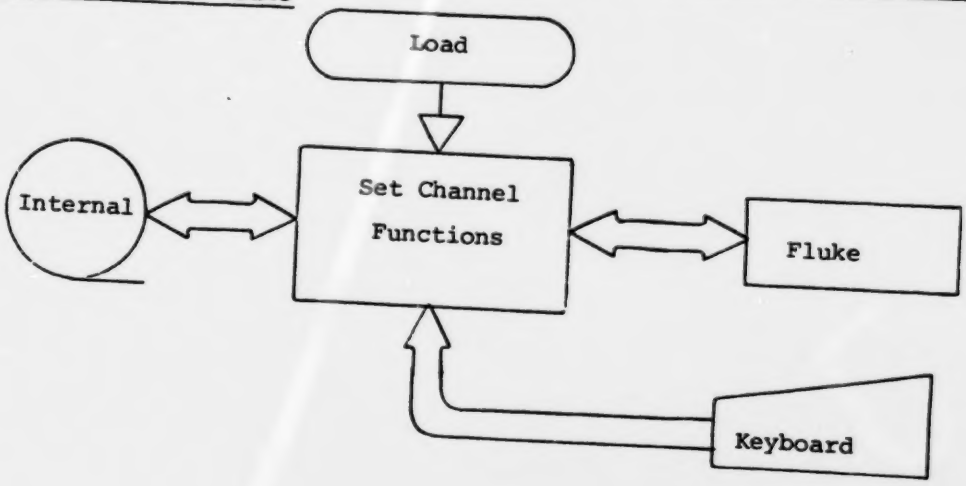
Device Select Codes for printer, display and internal tape drive are the standard ones specified in HP manuals.

## 3.7 System Flow Charts

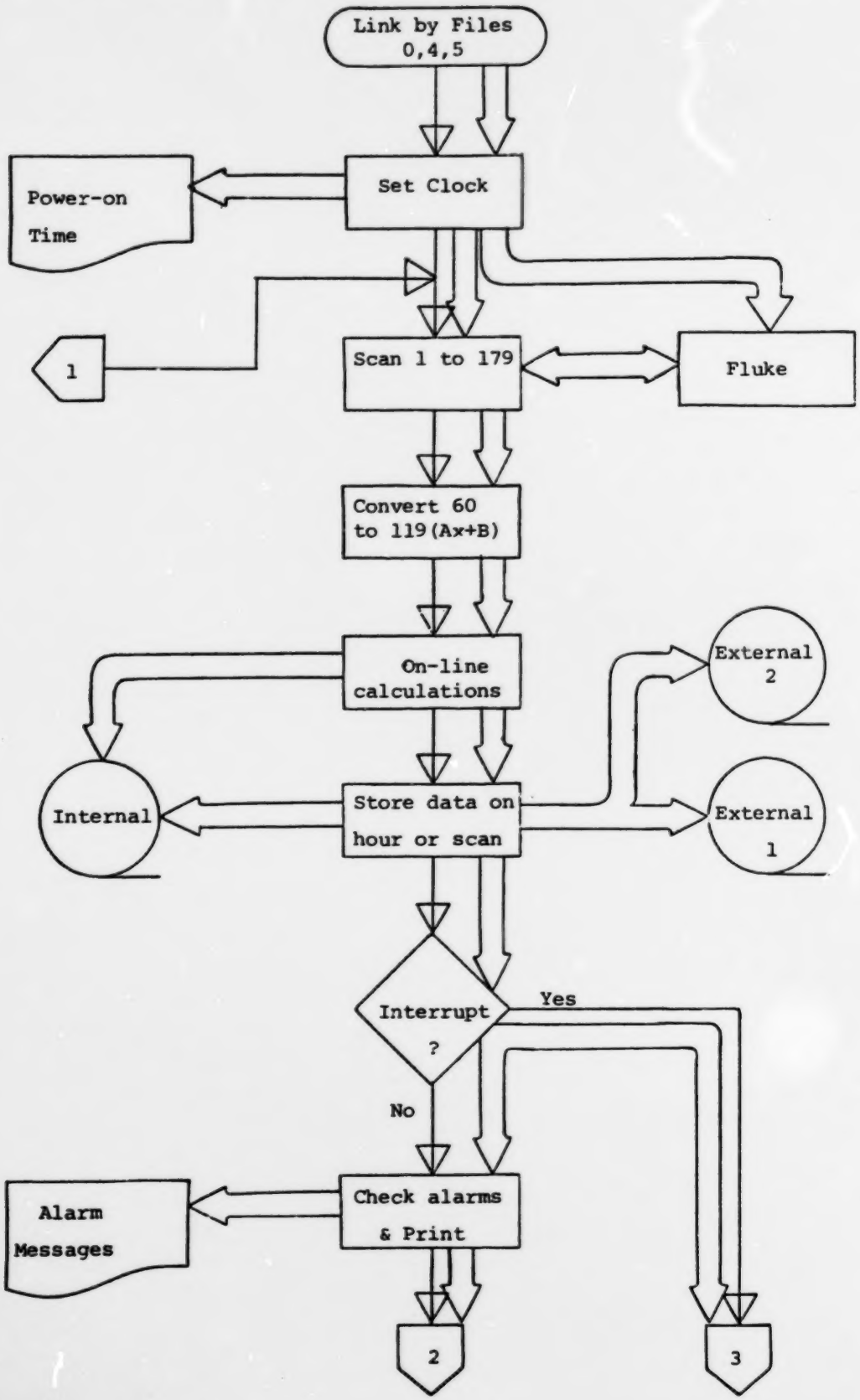
### File 0 - Automatic Start



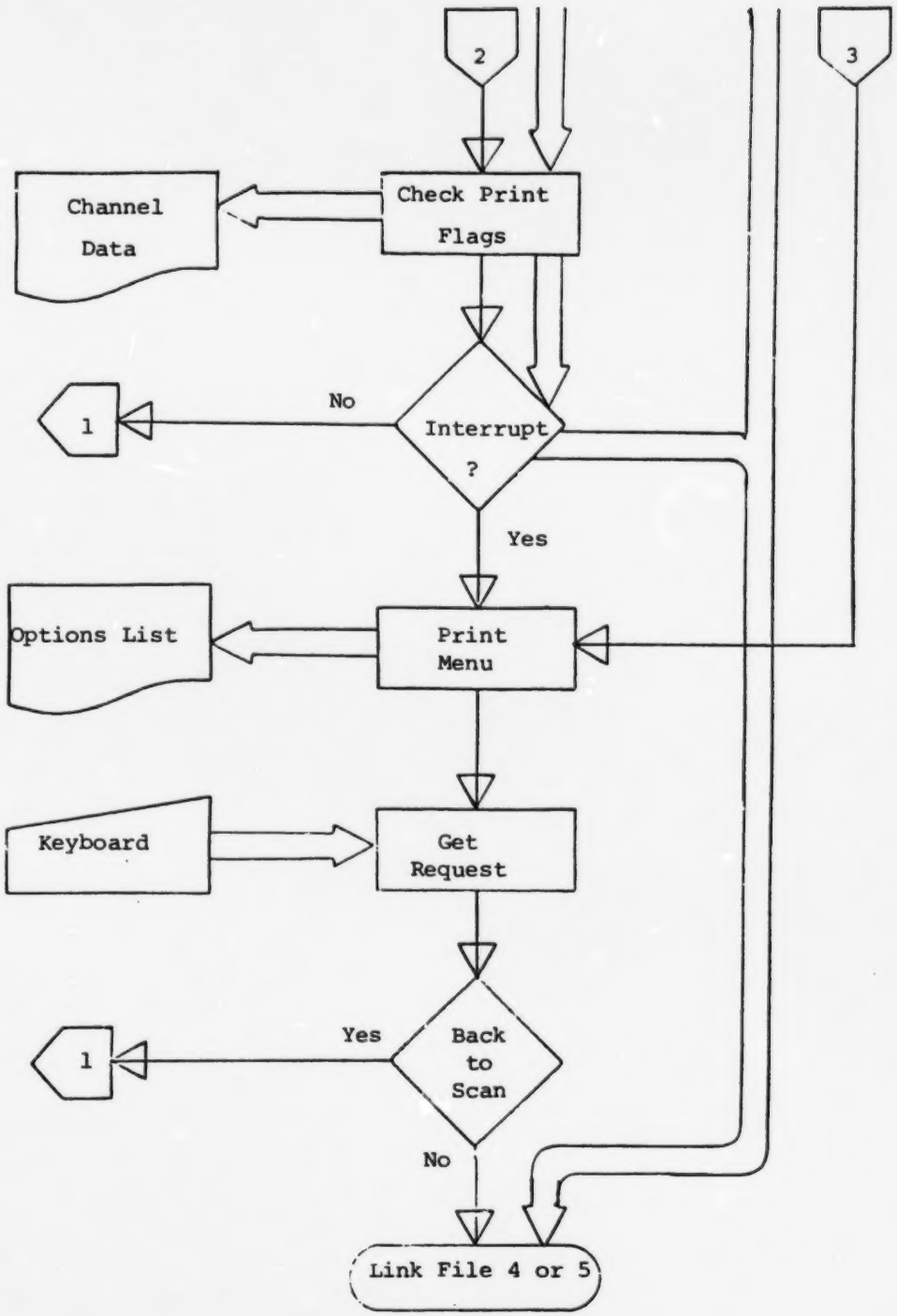

### File 6 - Program Fluke



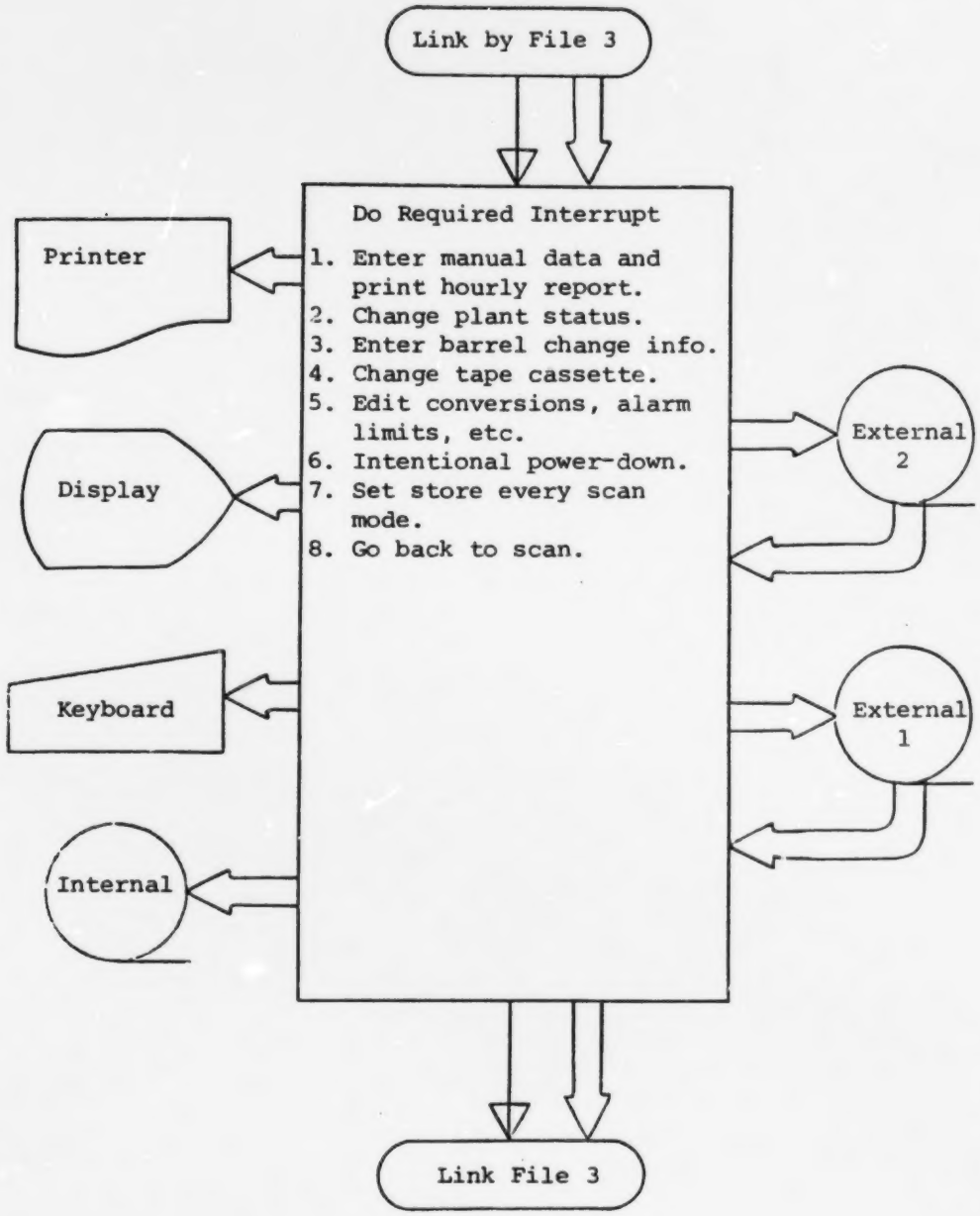
File 3 - Scan and Store



File 3 (cont'd)


2
3
Channel Data
Check Print Flags
1
No
Interrupt ?
Yes
Options List
Print Menu
Keyboard
Get Request
1
Yes
Back to Scan
No
Link File 4 or 5

## Files 4 and 5 - Interrupt Routines



# 4. Program Flow Charts and Source Listings

## 4.1 Introduction

In the following sections, a brief description of the uses of each program is followed by a flow chart and source listing. The source listings are commented (REM statements) as extensively as was possible within the fairly tight memory constraints imposed by the hardware and the use of the BASIC language.

The program documentation is broken down on a file-by-file basis with reference to the tape files on the system cassette tape. Individual subroutines within a file are not separately documented unless they are of sufficient length or complexity to make it difficult to understand their purpose or operation from the source code.

## 4.2 File 0 - Initialization and Power-on Start

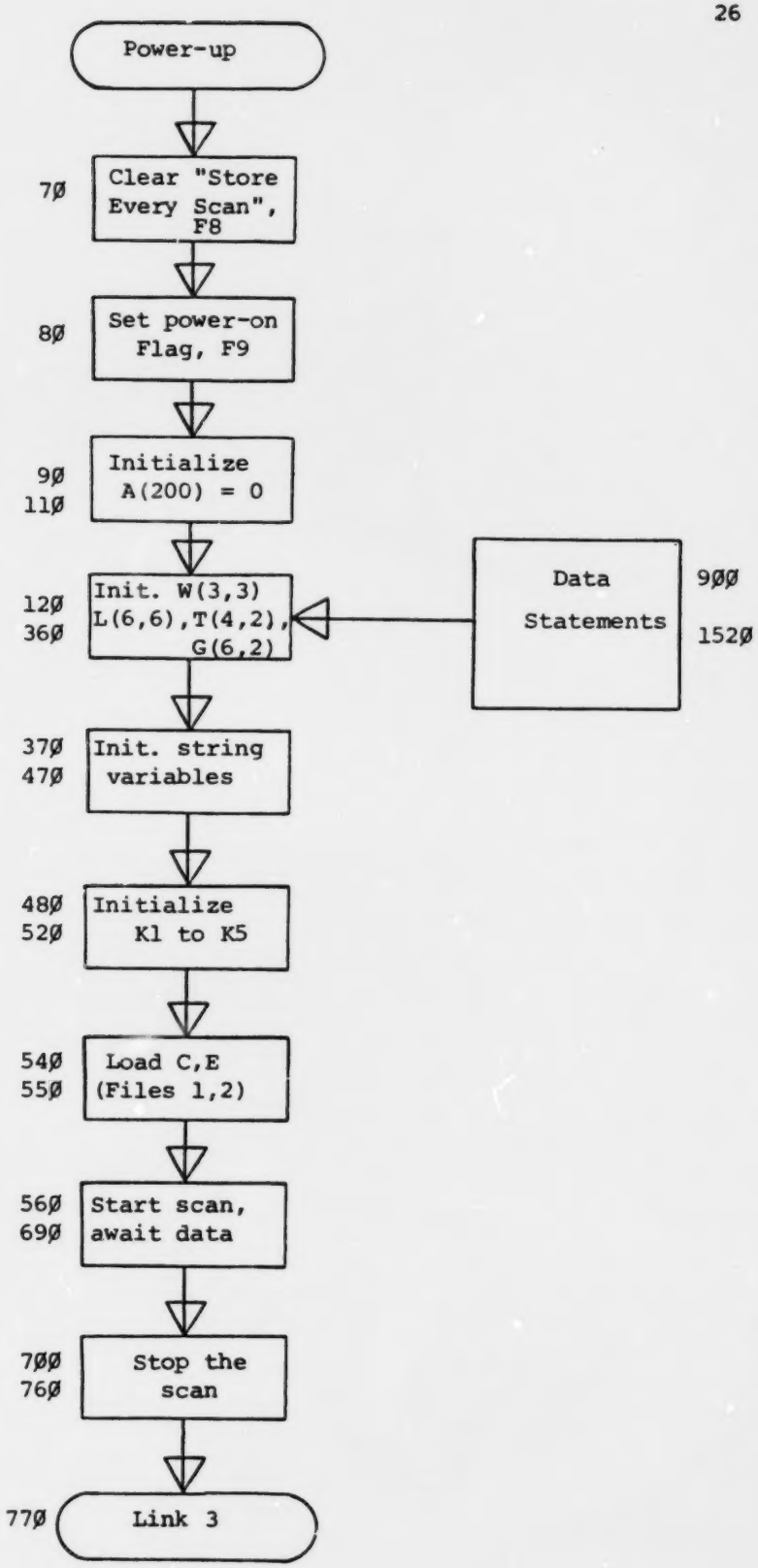
When power is switched on to the 9830A, a hardware provision causes internal tape File 0 to be loaded into memory and run if the internal tape cassette is in place.

This file initializes several tables from data statements in the program. The data statements which pertain to external factors such as data logger channel allocations are grouped at the end of the program for ease in changing them should it become necessary. Instructions on what calculations are performed and which channels are assumed to be which variables are contained in the program comments.

Several string variables, numerical variables and constants are also initialized from data within File 0. The program also reads the data contained in Files 1 and 2 into memory. These files contain all the current working information and flags required by the system. Whenever changes are made to this information in the course of operation, an updated version is stored in Files 1 and 2 so that recovery from a power failure is possible and so that it is not necessary to enter data such as alarm limits except when a change is desired.

At the point labelled "Power-on Start", a procedure is executed to ensure that the data logger is in a known operating state. The reasons for this are explained in Chapter 5 below.

When the system has been properly initialized, the program links File 3.


Power-up
7Ø
Clear "Store Every Scan", F8
8Ø
Set power-on Flag, F9
9Ø
11Ø
Initialize A(200) = 0
12Ø
36Ø
Init. W(3,3) L(6,6),T(4,2), G(6,2)
Data Statements
9ØØ
152Ø
37Ø
47Ø
Init. string variables
48Ø
52Ø
Initialize K1 to K5
54Ø
55Ø
Load C,E (Files 1,2)
56Ø
69Ø
Start scan, await data
7ØØ
76Ø
Stop the scan
77Ø
Link 3

```
10 COM P$[130],H$[20],N$[20],T$[80],U$[80],V$[80]
20 DIM A$[20],B$[20],C$[20],D$[35],E$[3],F$[15],G$[80],M$[60],P$[30],S$[3]
30 DIM W$[80],X$[20],Y$[20],Z$[20],A$[200],C$[200,3]
40 DIM D[6],E[200,3],F[2],G[5,2],L[6,6],T[4,2],W[3,3],X[6],Y[4]
50 REM----INITIALIZATION----------------------------------------------------
60 FIND 1
70 F8=1
80 F9=0
90 FOR I=1 TO 200
100 A[I]=0
110 NEXT I
120 RESTORE 180
130 FOR I=1 TO 3
140 FOR J=1 TO 3
150 READ W[I,J]
160 NEXT J
170 NEXT I
180 DATA 3,1,1,2,3,3,2,3,3
190 RESTORE 940
200 FOR I=1 TO 6
210 FOR J=1 TO 6
220 READ L[I,J]
230 NEXT J
240 NEXT I
250 RESTORE 1200
260 FOR I=1 TO 4
270 FOR P=1 TO 2
280 READ T[I,P]
290 NEXT P
300 NEXT I
310 RESTORE 1340
320 FOR I=1 TO 5
330 FOR P=1 TO 2
340 READ G[I,P]
350 NEXT P
360 NEXT I
370 W$="DEG CMV    KG    KG/HRXH2  KPA   MPA   L/HR V     HTC   HTR          "
380 C$="M"
390 M$="JAN. FEB. MARCHAPRILMAY. JUNE JULY AUG. SEPT. OCT. NOV. DEC. "
400 D$="SAT. SUN. MON. TUE. WED. THUR. FRI. "
410 P$="RUNNING    ALARMS OFFSHUT DOWN "
420 G$="LIGHT END HEAVY END SCRUBBER  RECYCLE    "
430 G$[41]="REACTOR    PREHEATER X          Y          "
440 OUTPUT (S$,460)18
450 OUTPUT (E$,460)20
460 FORMAT B
470 X$="E1E1W4X0S1"
480 K1=1.4907
490 K2=0.1245
500 K3=1
510 K4=0.874
520 K5=0.7842
```

```
530 REM---POWER ON START-----------------------------------------------
540 LOAD  DATA 1,C
550 LOAD  DATA 2,E
560 FOR I=1 TO 7 STEP 2
570 A$=X$[I,I+1]
580 GOSUB 780
590 NEXT I
600 X$="M001F001L179"
610 FOR I=1 TO 9 STEP 4
620 A$=X$[I,I+3]
630 GOSUB 780
640 NEXT I
650 A$="S1"
660 GOSUB 780
670 ENTER (9,*)A$
680 IF A$[1,1]="@" THEN 560
690 IF A$[1,1]#"A" THEN 670
700 A$="S0"
710 GOSUB 780
720 GOSUB 830
730 A$="F001"
740 GOSUB 780
750 A$="L179"
760 GOSUB 780
770 LINK 3,10,50
780 REM---COMMAND SENDING SUBROUTINE-----------------------------------
790 OUTPUT (9,*)A$;
800 OUTPUT (9,460)13;
810 ENTER (9,*)B$
820 RETURN
830 REM---WAIT FOR END OF SCAN-----------------------------------------
840 ENTER (9,*)B$
850 IF B$[1,1]#E$[1,1] THEN 840
860 RETURN
870 REM
880 REM
890 REM
```

```
900 REM-----DATA TABLES L,T&G FOR MATCHING CHANNEL NUMBER TO DATA LOCATION----
910 REM
920 REM
930 REM
940 REM---LOCATIONS OF HEAT TRANSFER DATA IN A(200)---
950 REM  HTC=K2*(T1-(T2+T3)/2)/(T4-T1)
960 REM  HTR=K1*(T1-(T2+T3)/2)
970 REM T1,T2,T3,T4,HTC,HTR (LOCATIONS IN A(200))
980 REM
990 REM---PLANT 1---
1000 REM
1010 REM 5
1020 DATA 16,56,57,15,190,193
1030 REM 4
1040 DATA 14,18,17,13,191,194
1050 REM 2
1060 DATA 55,20,19,11,192,195
1070 REM
1080 REM---PLANT 2---
1090 REM
1100 REM 5
1110 DATA 121,122,123,124,180,183
1120 REM 4
1130 DATA 125,126,127,128,181,184
1140 REM 2
1150 DATA 129,130,131,132,182,185
1160 REM
1170 REM
1180 REM
1190 REM
1200 REM---LOCATIONS OF FEED TANK DATA IN A(200)---
1210 REM
1220 REM
1230 REM
1240 REM IN THE FORM---DATA (PLANT 1),(PLANT 2)---
1250 REM
1260 REM FEED TANK 1
1270 DATA 63,91
1280 REM FEED TANK 2
1290 DATA 62,92
1300 REM FEED TANK SWITCH
1310 DATA 61,93
1320 REM FEED RATE
1330 DATA 198,188
1340 REM
1350 REM
1360 REM
```

```
1370 REM---RECYCLE GAS RATE DATA LOCATIONS---
1380 REM
1390 REM
1400 REM    G=K3*SQR(PD)/(K4-K5*X)*SQR(P))
1410 REM    G(CORRECTED FOR TEMP)=G*287/(273+TEMP)
1420 REM
1430 REM---GAS RATE PD---
1440 DATA 80,133
1450 REM---HYDROGEN PRESSURE---
1460 DATA 86,133
1470 REM---MOLE % HYDROGEN---
1480 DATA 64,133
1490 REM---AMBIENT TEMPERATURE---
1500 DATA 51,133
1510 REM---RECYCLE GAS RATE---
1520 DATA 197,187
```

## 4.3 File 3 - Scanning and Data Storage

This program is the normal running program for the system and carries out the major functions of collecting data from the Fluke data logger, scaling some of the values, doing some on-line calculations, storing of data every hour, checking for out of range conditions on those channels for which this is desirable, printing out data from specified channels so that they can be monitored and running the interrupt handling routine which allows the operator to communicate with the system when some operating change is occurring or when manually obtained data is to be entered for storage with the scan data.

When an out-of-range condition is encountered on a channel, the program does not immediately print the alarm. Since the possibility exists of false alarms being generated due to noise in the system, the program asks the data logger to take one or more additional readings from the channel in question. The channel is read three times in succession unless a further out-of-range reading is obtained. In this latter case, the alarm message is printed. If the additional readings are not out-of-range then the last value read is stored as the correct value for the channel in place of the original, erroneous reading.

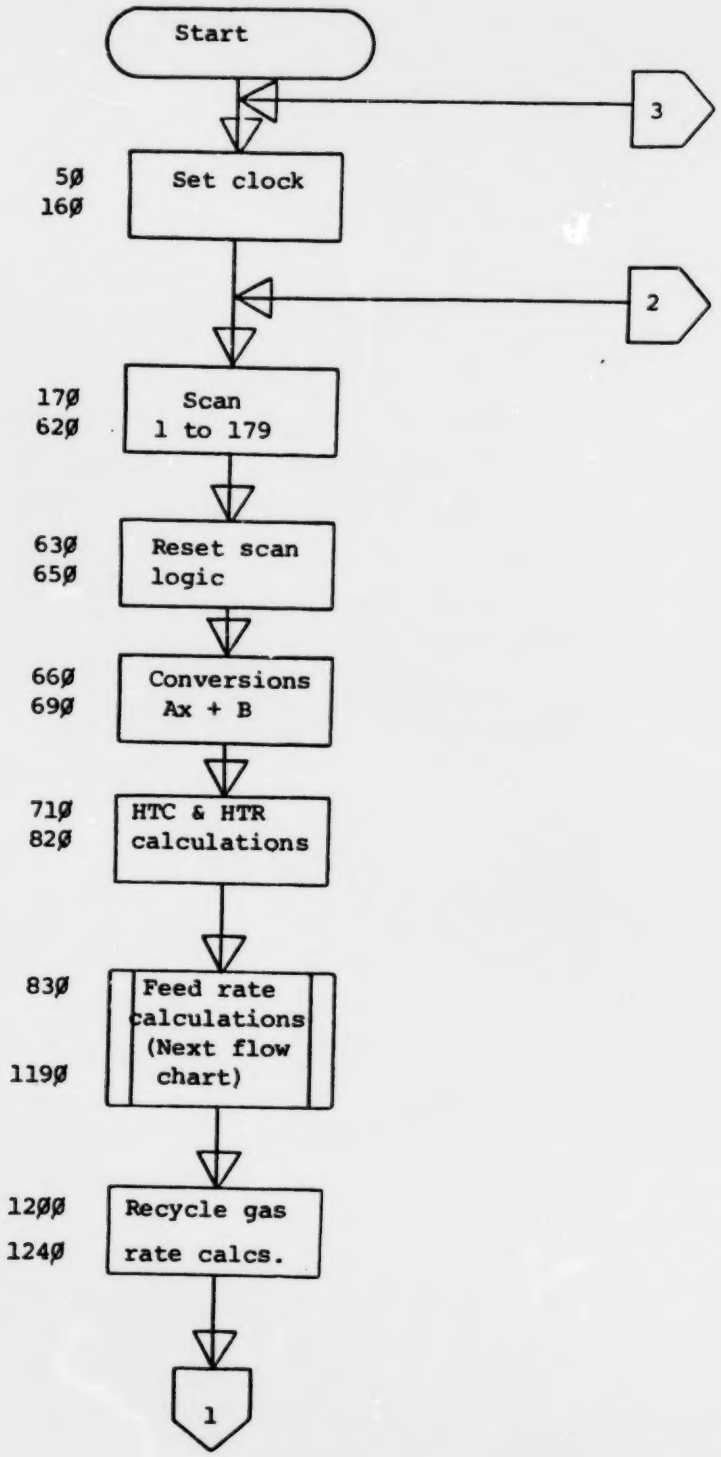
In the case where an out-of-range value is obtained on one of the channels containing calculated values (i.e. Channels 180 to 200) then the alarm is printed immediately without a confirmation.

If the interrupt switch is activated when there are less than 10 minutes before a change in the hour, then a warning to this effect is printed out. This is because when the system is servicing the interrupt and carrying out any of the functions requiring operator interaction, it cannot, simultaneously, continue to scan and store data. Thus, there is a danger of missing the

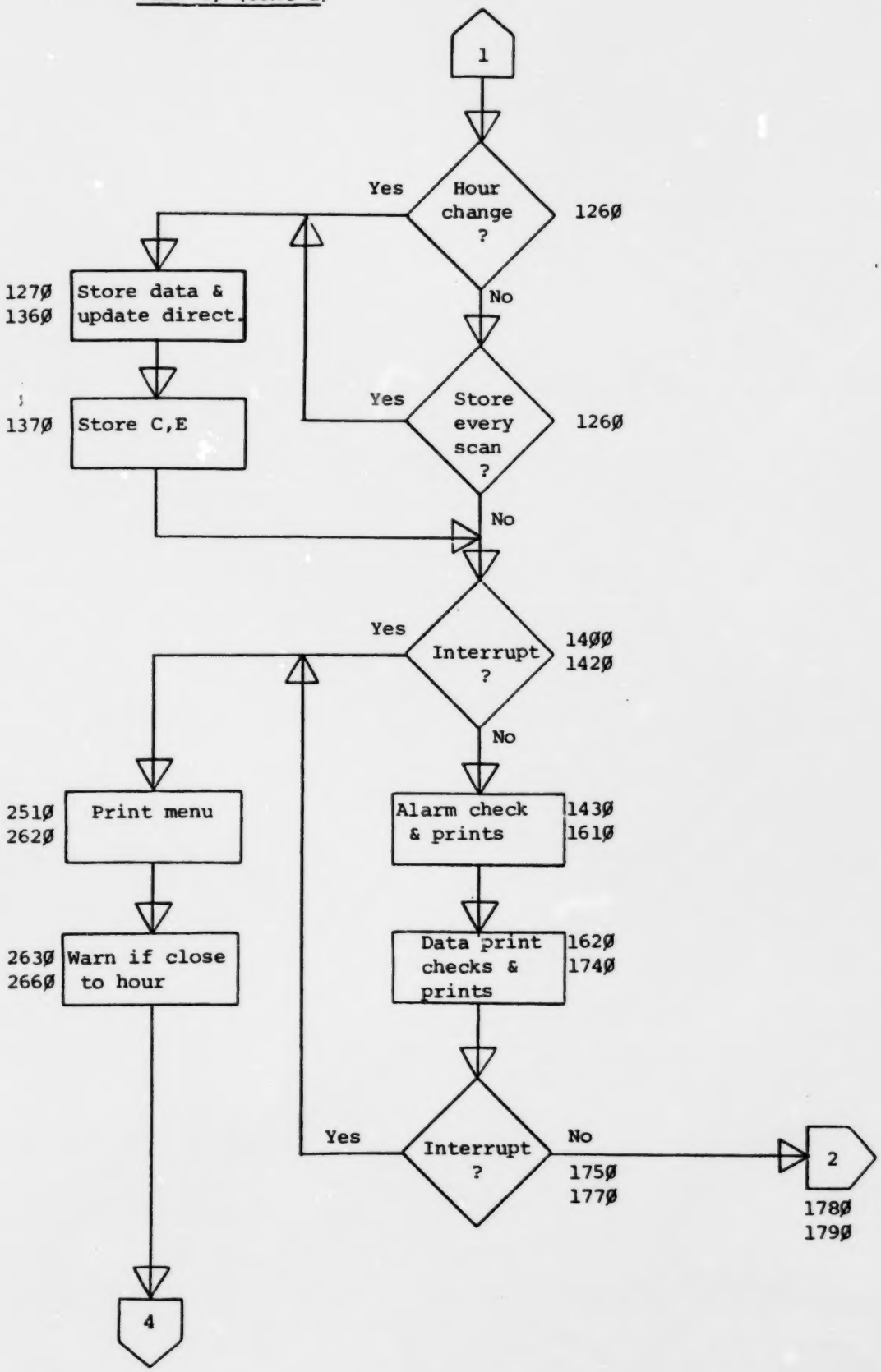
data for the first scan after the hour. Of course, the first scan completed after the interrupt is complete and after the hour, will still be stored.

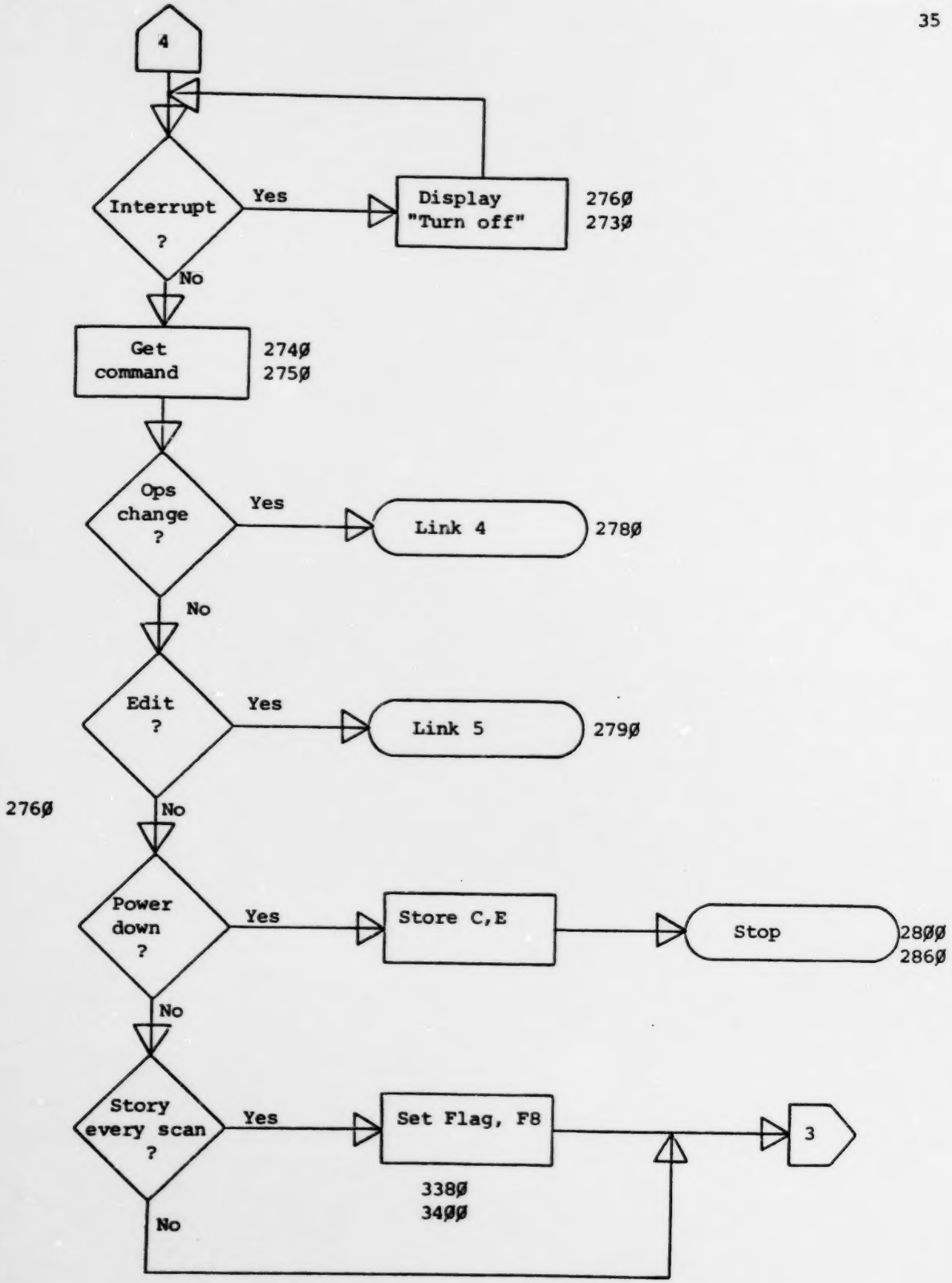
The interrupt switch is connected to one channel (Channel 0) of the data logger. Thus, activation of the switch produces a response only after completion of the next scan. In order to ensure that the interrupt switch is not inadvertently left on, the interrupt handling segment of the program prints a message to the operator asking that the interrupt switch be turned off. Processing of the interrupt will not procede until this has been done.

File 3 - Scanning and Data Storage

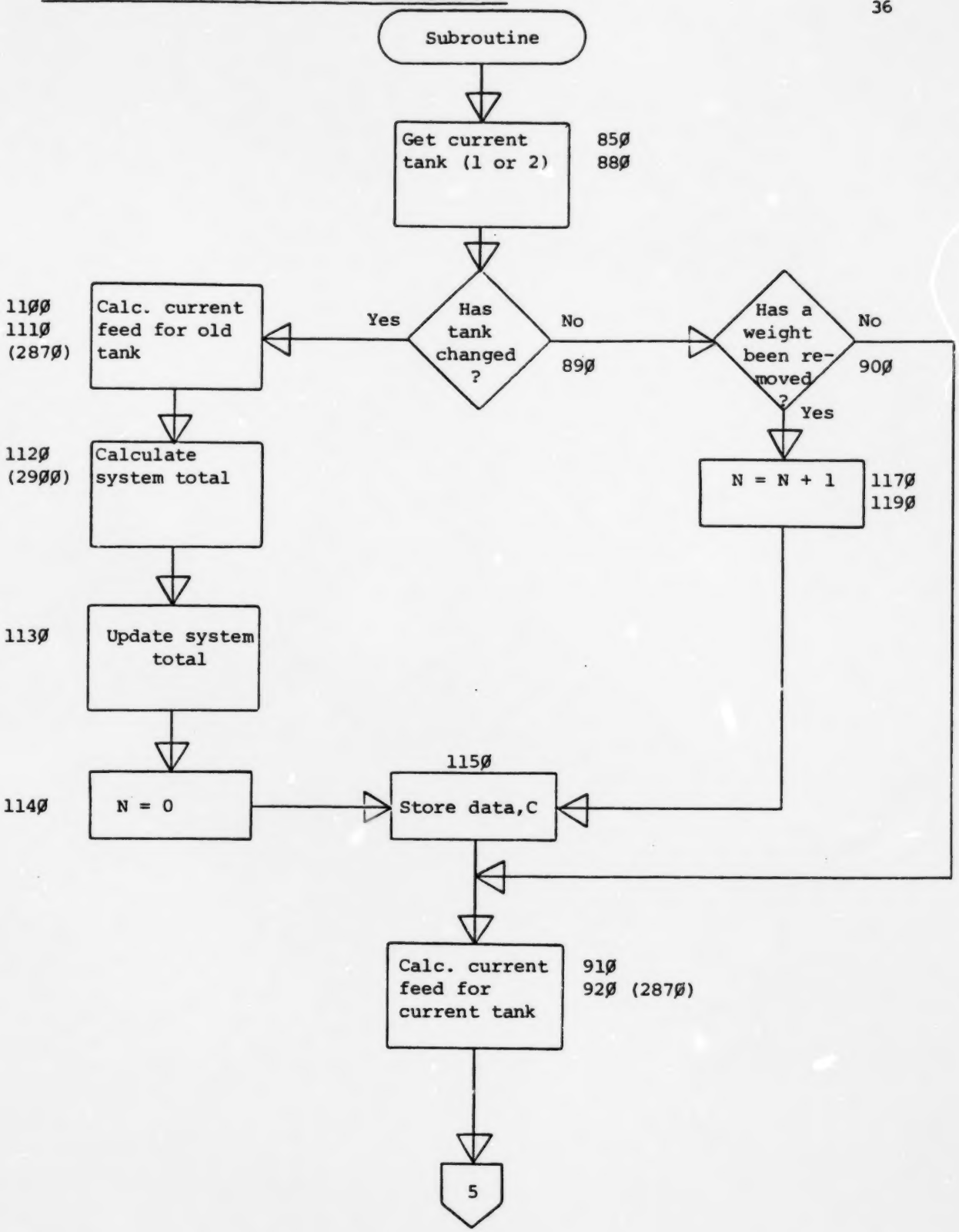


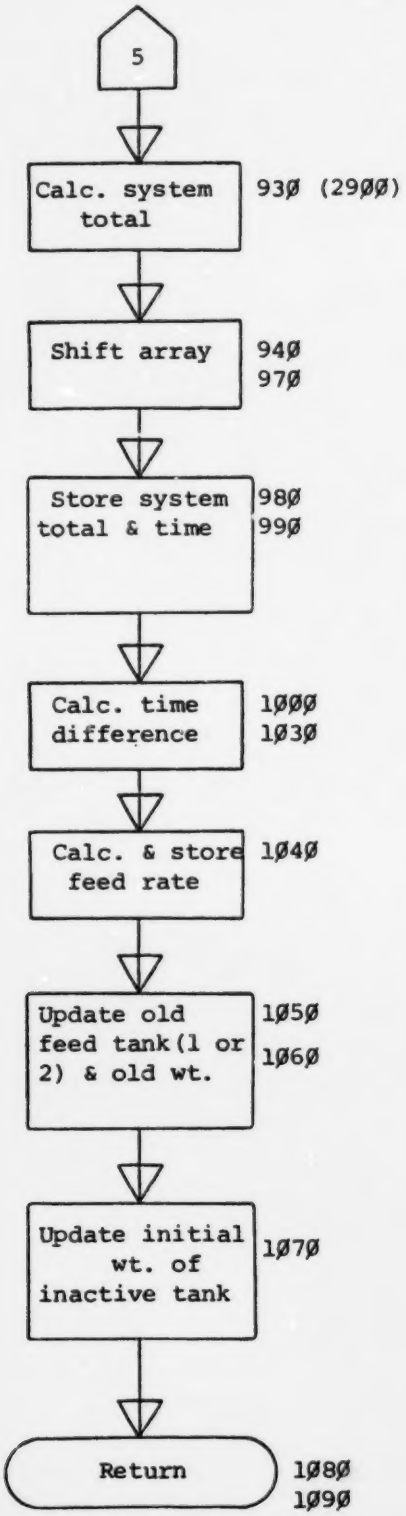
File 3, (cont'd)


1
Yes
Hour change ?
126Ø
No
127Ø
136Ø
Store data & update direct.
137Ø
Store C,E
Yes
Store every scan ?
126Ø
No
Yes
Interrupt ?
14ØØ
142Ø
No
251Ø
262Ø
Print menu
263Ø
266Ø
Warn if close to hour
4
Alarm check & prints
143Ø
161Ø
Data print checks & prints
162Ø
174Ø
Yes
Interrupt ?
No
175Ø
177Ø
2
178Ø
179Ø


4
Interrupt ?
Yes
Display "Turn off"
276Ø
273Ø
No
Get command
274Ø
275Ø
Ops change ?
Yes
Link 4
278Ø
No
Edit ?
Yes
Link 5
279Ø
276Ø
No
Power down ?
Yes
Store C,E
Stop
28ØØ
286Ø
No
Story every scan ?
Yes
Set Flag, F8
3
338Ø
34ØØ
No

File 3, (cont'd) - Feed Rate Calculations




5
Calc. system total
93Ø (29ØØ)
Shift array
94Ø
97Ø
Store system total & time
98Ø
99Ø
Calc. time difference
1ØØØ
1Ø3Ø
Calc. & store feed rate
1Ø4Ø
Update old feed tank(1 or 2) & old wt.
1Ø5Ø
1Ø6Ø
Update initial wt. of inactive tank
1Ø7Ø
Return
1Ø8Ø
1Ø9Ø

```
10 COM P$[130],H$[20],N$[20],T$[80],U$[80],V$[80]
20 DIM A$[20],B$[20],C$[20],D$[35],E$[3],F$[15],G$[80],M$[60],P$[30],S$[3]
30 DIM W$[80],X$[20],Y$[20],Z$[20],A$[200],C$[200,3]
40 DIM D[6],E[200,3],F[2],G[5,2],L[6,6],T[4,2],W[3,3],X[6],Y[4]
50 REM---RE-ENTER HERE FROM INTERRUPT---
60 FIND 4
70 A$="W4"
80 GOSUB 2300
90 A$="S1"
100 GOSUB 2300
110 ENTER (9,*)A$
120 IF A$[1,1]="D" THEN 360
130 IF A$[1,1]#"A" THEN 110
140 A$="S0"
150 GOSUB 2300
160 GOSUB 2360
170 REM---DO A SCAN----------------------------------------------------------
180 A$="W4"
190 GOSUB 2300
200 A$="F001"
210 GOSUB 2300
220 A$="L179"
230 GOSUB 2300
240 A$="S1"
250 GOSUB 2300
260 ENTER (9,*)A$
270 IF A$[1,1]="A" THEN 390
280 IF A$[1,1]="Y" THEN 420
290 IF A$[1,1]="X" THEN 500
300 IF A$[1,1]="D" THEN 260
310 IF A$[1,1]=S$[1,1] THEN 260
320 IF A$[1,1]=E$[1,1] THEN 640
330 PRINT "BAD DATA FROM FLUKE ERROR"
340 PRINT A$
350 GOTO 260
360 GOSUB 580
370 GOSUB 2390
380 GOTO 110
390 REM GET ANALOG DATA-----------------------------------------------------
400 A[VAL(A$[2,4])]=VAL(A$[6,12])
410 GOTO 260
420 REM---GET TIME-----------------------------------------------------------
430 Y[1]=VAL(A$[2,4])
440 Y[2]=VAL(A$[6,7])
450 Y[3]=VAL(A$[9,10])
460 Y[4]=VAL(A$[12,13])
470 H$=A$[2,13]
480 C[183,3]=Y[2]*3600+Y[3]*60+Y[4]
490 GOTO 260
```

```
500 REM---GET FIXED DATA-----------------------------------------------
510 FOR I=1 TO 6
520 X[I]=VAL(A$[I+1,I+1])
530 NEXT I
540 GOTO 260
550 REM---GET DIGITAL DATA (CLOCK)-------------------------------------
560 GOSUB 580
570 GOTO 260
580 REM
590 D[1]=VAL(A$[6,7])
600 D[2]=VAL(A$[9,10])
610 D[3]=VAL(A$[12,13])
620 RETURN
630 REM---SCAN FINISHED-----------------------------------------------
640 A$="SO"
650 GOSUB 2300
660 REM---DO CONVERSIONS-----------------------------------------------
670 FOR I=1 TO 60
680 A[I+59]=A[I+59]*C[I,3]+C[I+60,3]
690 NEXT I
700 REM----ON LINE CALCULATIONS---------------------------------------
710 REM---HTC AND HTR CALCULATIONS--------------
720 FOR I=1 TO 6
730 Z1=A[L[I,4]]-A[L[I,1]]
740 IF Z1=0 THEN 790
750 IF ABS(Z1)<0.1 THEN 810
760 Z=A[L[I,1]]-(A[L[I,2]]+A[L[I,3]])/2
770 A[L[I,5]]=K2*Z/Z1
780 A[L[I,6]]=K1*Z
790 NEXT I
800 GOTO 830
810 Z1=0.1*SGN(Z1)
820 GOTO 760
```

```
830 REM---CALCULATE FEED RATE-----------------------------------------------
840 FOR P=1 TO 2
850 S=0
860 IF A[T[3,P]]>1.5 THEN 880
870 S=1
880 S0=ABS(S-1)
890 IF S#C[124+(P-1)*30,3] THEN 1100
900 IF A[T[1+S,P]]>5+C[125+(P-1)*30,3] THEN 1170
910 X=S
920 GOSUB 2870
930 GOSUB 2900
940 FOR I=0 TO 6 STEP 2
950 C[143+I+(P-1)*30,3]=C[143+I+2+(P-1)*30,3]
960 C[143+I+1+(P-1)*30,3]=C[143+I+3+(P-1)*30,3]
970 NEXT I
980 C[143+(P-1)*30+6,3]=T
990 C[143+(P-1)*30+7,3]=C[183,3]
1000 T=C[143+7+(P-1)*30,3]-C[143+1+(P-1)*30,3]
1010 T=T/3600
1020 IF T>0 THEN 1040
1030 T=T+24
1040 A[T[4,P]]=(C[143+6+(P-1)*30,3]-C[143+(P-1)*30,3])/T
1050 C[124+(P-1)*30,3]=S
1060 C[125+(P-1)*30,3]=A[T[1+S,P]]
1070 C[121+S0+(P-1)*30,3]=A[T[1+S0,P]]
1080 NEXT P
1090 GOTO 1200
1100 X=S0
1110 GOSUB 2870
1120 GOSUB 2900
1130 C[126+(P-1)*30,3]=T
1140 C[123+(P-1)*30,3]=0
1150 STORE  DATA 1,C
1160 GOTO 910
1170 C[123+(P-1)*30,3]=C[123+(P-1)*30,3]+1
1180 STORE  DATA 1,C
1190 GOTO 910
```

```
1200 REM---RECYCLE GAS RATE---
1210 FOR P=1 TO 2
1220 G=K3*SQR(ABS(A[G[1,P]]))/(K4-K5*A[G[3,P]])*SQR(ABS(A[G[2,P]]))
1230 A[G[5,P]]=G*287/(273+A[G[4,P]])
1240 NEXT P
1250 REM---CHECK FOR HOUR CHANGE---
1260 IF Y[2]=C[186,3] AND F8 THEN 1400
1270 IF Y[1] <= C[187,3] THEN 1320
1280 Y[1]=1
1290 C[182]=1
1300 C[181,3]=C[181,3]+1
1310 GOTO 50
1320 FOR P=1 TO 2
1330 GOSUB 3020
1340 NEXT P
1350 C[186,3]=Y[2]
1360 C[182,3]=Y[1]
1370 STORE  DATA 1,C
1380 STORE  DATA 2,E
1390 FIND 4
1400 REM---CHECK FOR INTERRUPT---
1410 GOSUB 2930
1420 IF F5 THEN 2510
1430 REM---CHECK FOR ALARM CONDITION----------------------------------------
1440 GOTO 1490
1450 FOR I=N1 TO N2
1460 IF A[I]<C[I,1]-C[I,2] OR A[I]>C[I,1]+C[I,2] THEN 1800
1470 NEXT I
1480 RETURN
1490 F1=1
1500 IF C[184,3] THEN 1540
1510 N1=1
1520 N2=89
1530 GOSUB 1450
1540 IF C[185,3] THEN 1580
1550 N1=90
1560 N2=189
1570 GOSUB 1450
1580 IF C[184,3] THEN 1620
1590 N1=190
1600 N2=200
1610 GOSUB 1450
1620 REM---CHECK FOR PRINT FLAGS---
1630 FOR I=1 TO 200
1640 IF E[I,2] THEN 1670
1650 NEXT I
1660 GOTO 1750
1670 REM PRINT FLAG MESSAGE-------------------------------------------------
1680 IF F1 THEN 1730
1690 PRINT "CHAN.  "; I
1700 PRINT "READS "A[I]"SHOULD BE "C[I,1]"+-"C[I,2]; W$[E[I,1]*5+1,E[I,1]*5+5]
1710 PRINT
1720 GOTO 1650
1730 GOSUB 2120
1740 GOTO 1690
```

File 3

```
1750 REM ---CHECK FOR INTERUPT----------
1760 GOSUB 2930
1770 IF F5 THEN 2510
1780 REM ---DO ANOTHER SCAN----------
1790 GOTO 170
1800 REM---DOUBLE CHECK FOR ALARM----------
1810 IF I>179 THEN 2230
1820 C1=I+1000
1830 OUTPUT (A$,*)C1
1840 C$[2]=A$[3,5]
1850 A$=C$
1860 GOSUB 2300
1870 A$="W4"
1880 GOSUB 2300
1890 A$="S3"
1900 GOSUB 2300
1910 J=1
1920 ENTER (9,*)A$
1930 IF A$[1,1]="A" THEN 1990
1940 GOTO 1920
1950 A$="S0"
1960 GOSUB 2300
1970 GOSUB 2360
1980 GOTO 1470
1990 A[I]=VAL(A$[6,12])
2000 IF A[I]<C[I,1]-C[I,2] OR A[I]>C[I,1]+C[I,2] THEN 2260
2010 J=J+1
2020 IF J=4 THEN 1950
2030 GOTO 1920
2040 REM---ALARM MESSAGE----------
2050 IF F1 THEN 2100
2060 PRINT "***ALARM ON CHAN. "I"***"
2070 PRINT "READS "A[I]"SHOULD BE "C[I,1]"+-"C[I,2];W$[E[I,1]*5+1,E[I,1]*5+5]
2080 PRINT
2090 RETURN
2100 GOSUB 2120
2110 GOTO 2060
2120 REM---PRINT TIME AND DATE----------
2130 PRINT
2140 WRITE (15,2150)
2150 FORMAT 75"*"
2160 WRITE (15,2170)H$;
2170 FORMAT 3"*",20X,"DATE AND TIME",2X,F4.0
2180 WRITE (15,2190)
2190 FORMAT 22X,3"*"
2200 WRITE (15,2150)
2210 F1=0
2220 RETURN
2230 REM---ALARM ON NON-SCAN DATA----------
2240 GOSUB 2050
2250 GOTO 1470
2260 REM---ALARM ON SCAN DATA----------
2270 GOSUB 2050
2280 GOTO 1950
```

```
2290 REM---COMMAND SENDING SUBROUTINE----------------------------------------
2300 OUTPUT (9,*)A$;
2310 OUTPUT (9,2340)13;
2320 ENTER (9,*)B$
2330 RETURN
2340 FORMAT B
2350 REM---WAIT FOR END OF SCAN----------------------------------------------
2360 ENTER (9,*)B$
2370 IF B$[1,1]#E$[1,1] THEN 2360
2380 RETURN
2390 REM---SET CLOCK---------------------------------------------------------
2400 D=1000000+10000*D[1]+100*D[2]+D[3]
2410 OUTPUT (T$,*)D;
2420 OUTPUT (A$,*)C[182,3]+1000
2430 A$[2]=A$[3]
2440 A$[5]=T$[3]
2450 A$[1,1]="T"
2460 GOSUB 2290
2470 IF F9 THEN 2500
2480 PRINT "POWER ON START AT"D[1]":"D[2]":"D[3]
2490 F9=1
2500 RETURN
2510 REM---INTERUPT SERVICE ROUTINE------------------------------------------
2520 GOSUB 2120
2530 PRINT "INTERUPT COMMAND LIST:"
2540 PRINT "1.    ENTER MANUAL DATA AND PRINT HOURLY REPORT"
2550 PRINT "2.    CHANGE PLANT STATUS"
2560 PRINT "3.    ENTER BARREL CHANGE INFO."
2570 PRINT "4.    CHANGE TAPE CASSETTE"
2580 PRINT "5.    EDIT CONVERSIONS,ALARM LIMITS ETC."
2590 PRINT "6.    INTENTIONAL POWER DOWN"
2600 PRINT "7.    SET STORE EVERY SCAN MODE"
2610 PRINT "8.    GO BACK TO SCAN"
2620 PRINT
2630 IF Y[3]<50 THEN 2670
2640 PRINT "********CAUTION*******"
2650 PRINT "LESS THAN"60-Y[3]"MIN.  TO HOUR"
2660 PRINT
2670 PRINT TAB(20);"TO CONTINUE TURN OFF INTERUPT"
2680 PRINT
2690 PRINT
2700 DISP "TURN OFF INTERUPT"
2710 WAIT 1000
2720 GOSUB 2930
2730 IF F5 THEN 2700
2740 DISP "ENTER COMMAND";
2750 INPUT W
2760 GOTO W OF 2780,2780,2780,2780,2790,2800,3380
2770 GOTO 50
2780 LINK 4,10,50
2790 LINK 5,10,50
```

```
2800 REM---INTENTIONAL POWER DOWN---
2810 C[184,3]=2
2820 C[185,3]=2
2830 STORE  DATA 1,C
2840 STORE  DATA 2,E
2850 PRINT "TURN OFF POWER"
2860 STOP
2870 REM---CURRENT FEED-----
2880 C[127+(P-1)*30,3]=C[121+X+(P-1)*30,3]-A[T[1+X,P]]+C[123+(P-1)*30,3]*10
2890 RETURN
2900 REM---SYSTEM TOTAL----
2910 T=C[126+(P-1)*30,3]+C[127+(P-1)*30,3]
2920 RETURN
2930 REM---CHECK INTERRUPT ---
2940 A$="R0004"
2950 GOSUB 2300
2960 IF B$[1,1]#"A" THEN 2940
2970 DISP
2980 F5=0
2990 IF VAL(B$[6,12])<1.5 THEN 3010
3000 F5=1
3010 RETURN
3020 REM---HOURLY DATA STORAGE---
3030 IF C[184+(P-1),3]>1 THEN 3110
3040 GOSUB 3120
3050 F=(P-1)*100+1
3060 IF E[F,3]=0 THEN 3090
3070 F=F+1
3080 GOTO 3060
3090 E[F,3]=Y[2]*1000+Y[1]
3100 STORE  DATA #5+(P-1),F-(P-1)*100
3110 RETURN
```

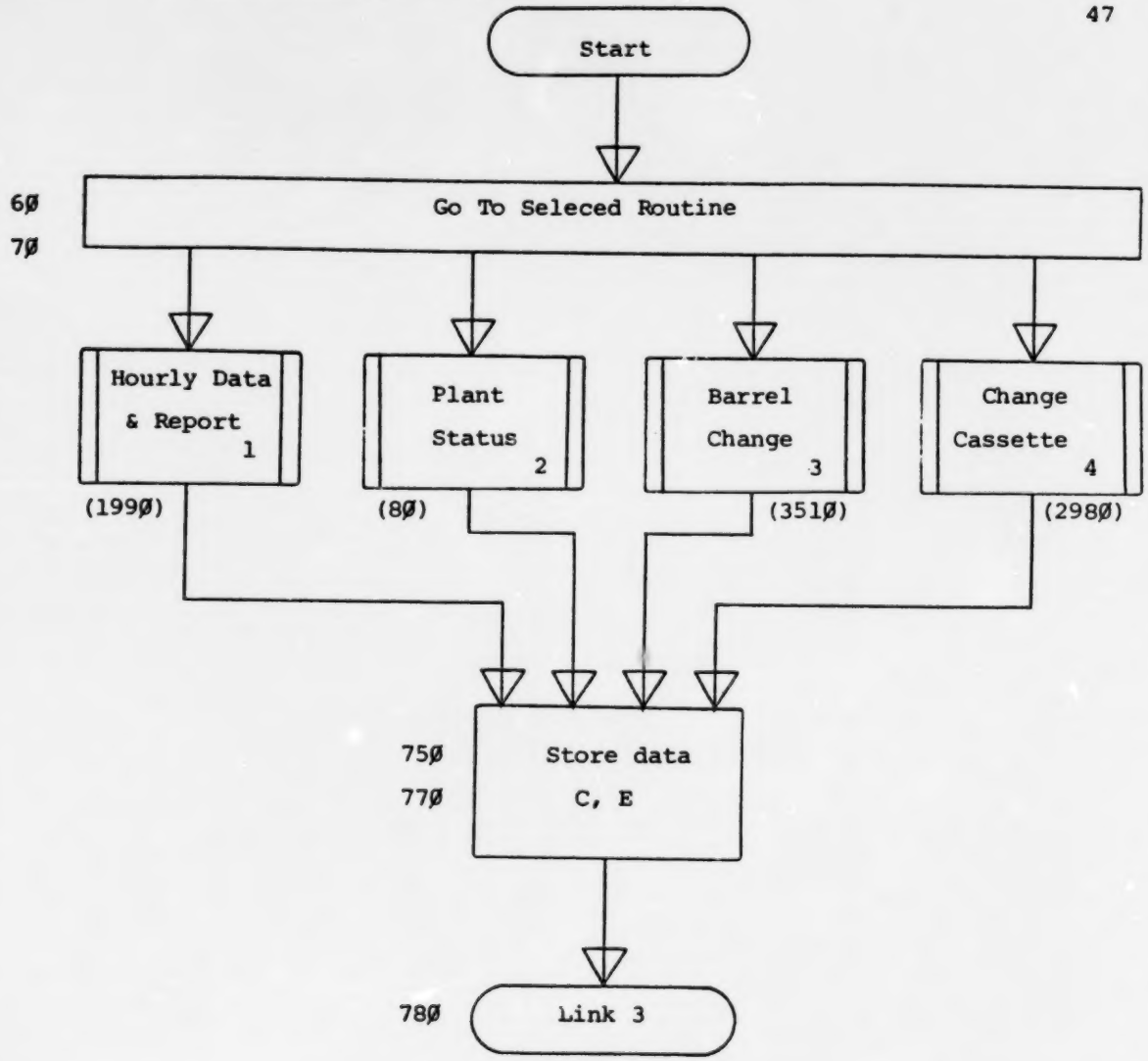
```
3120 REM---FILL UP P(130)---
3130 N$=Z$
3140 IF P=2 THEN 3160
3150 N$=Y$
3160 FOR I=1 TO 130
3170 P[I]=0
3180 NEXT I
3190 IF P=2 THEN 3240
3200 FOR I=1 TO 89
3210 P[I]=A[I]
3220 NEXT I
3230 IF P=1 THEN 3280
3240 FOR I=90 TO 189
3250 P[I-89]=A[I]
3260 NEXT I
3270 IF P=2 THEN 3310
3280 FOR I=190 TO 200
3290 P[I-100]=A[I]
3300 NEXT I
3310 FOR I=0 TO 15
3320 P[I+101]=C[121+I+(P-1)*30,3]
3330 NEXT I
3340 FOR I=0 TO 4
3350 P[I+126]=C[I+181,3]
3360 NEXT I
3370 RETURN
3380 REM---SET STORE EVERY SCAN---
3390 F8=0
3400 GOTO 170
```

## 4.4 File 4 - Operation Changes

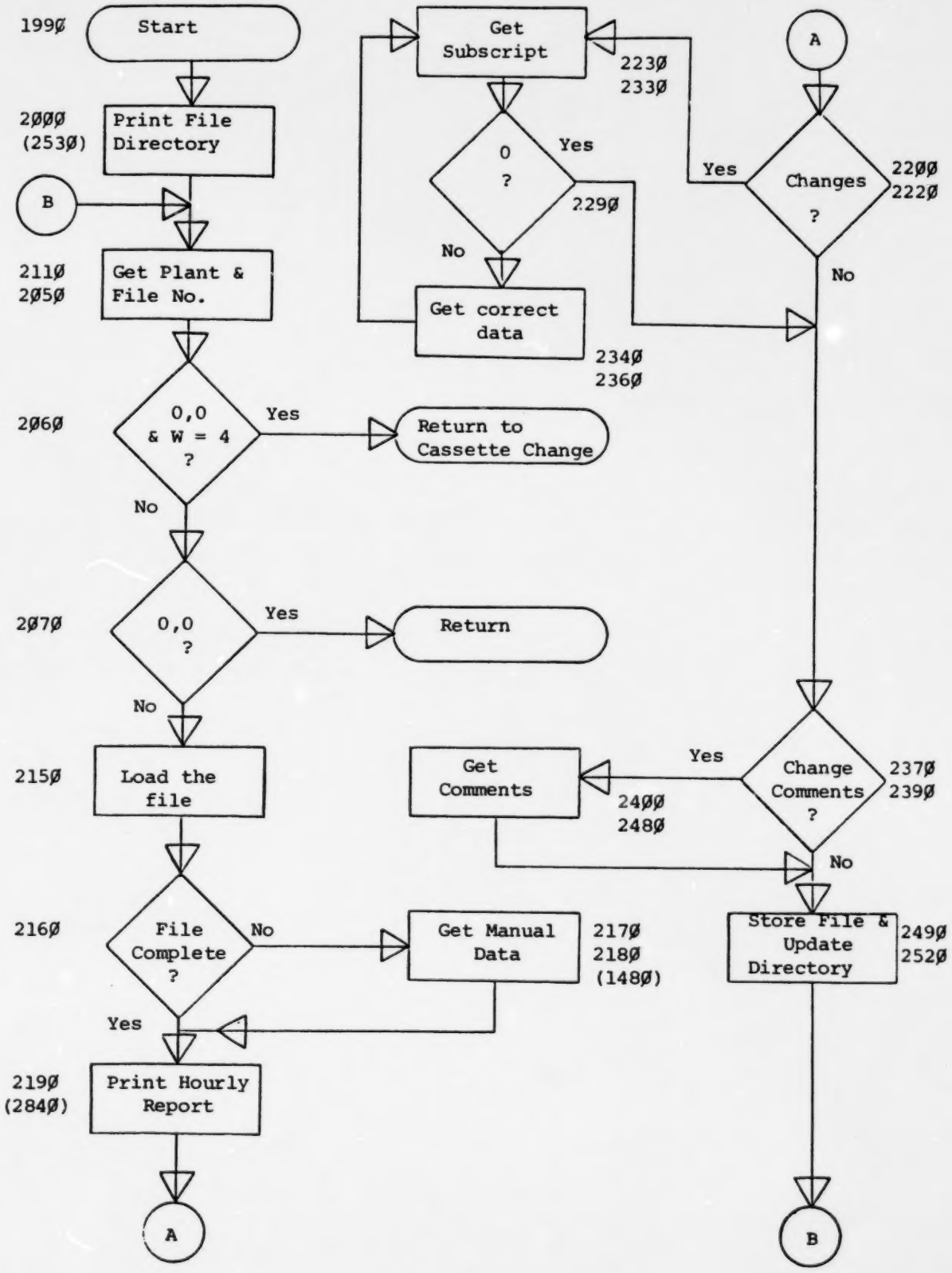
This program is linked into memory by the scanning program when the operator activates the interrupt switch and subsequently indicates a desire to enter manual data and produce an hourly report, enter a change in plant status, enter data pertaining to a change in a product barrel or change a data cassette.

Each of the subroutines is interactive and leads the operator through the necessary steps by asking for input of data values or text and by displaying appropriate menus of choices. The programs also handle all file management tasks required to protect data. If carelessly used, however, the "Change cassette" option has the potential to effectively destroy stored hourly data. The necessary precautions are discussed in the system operation manual.

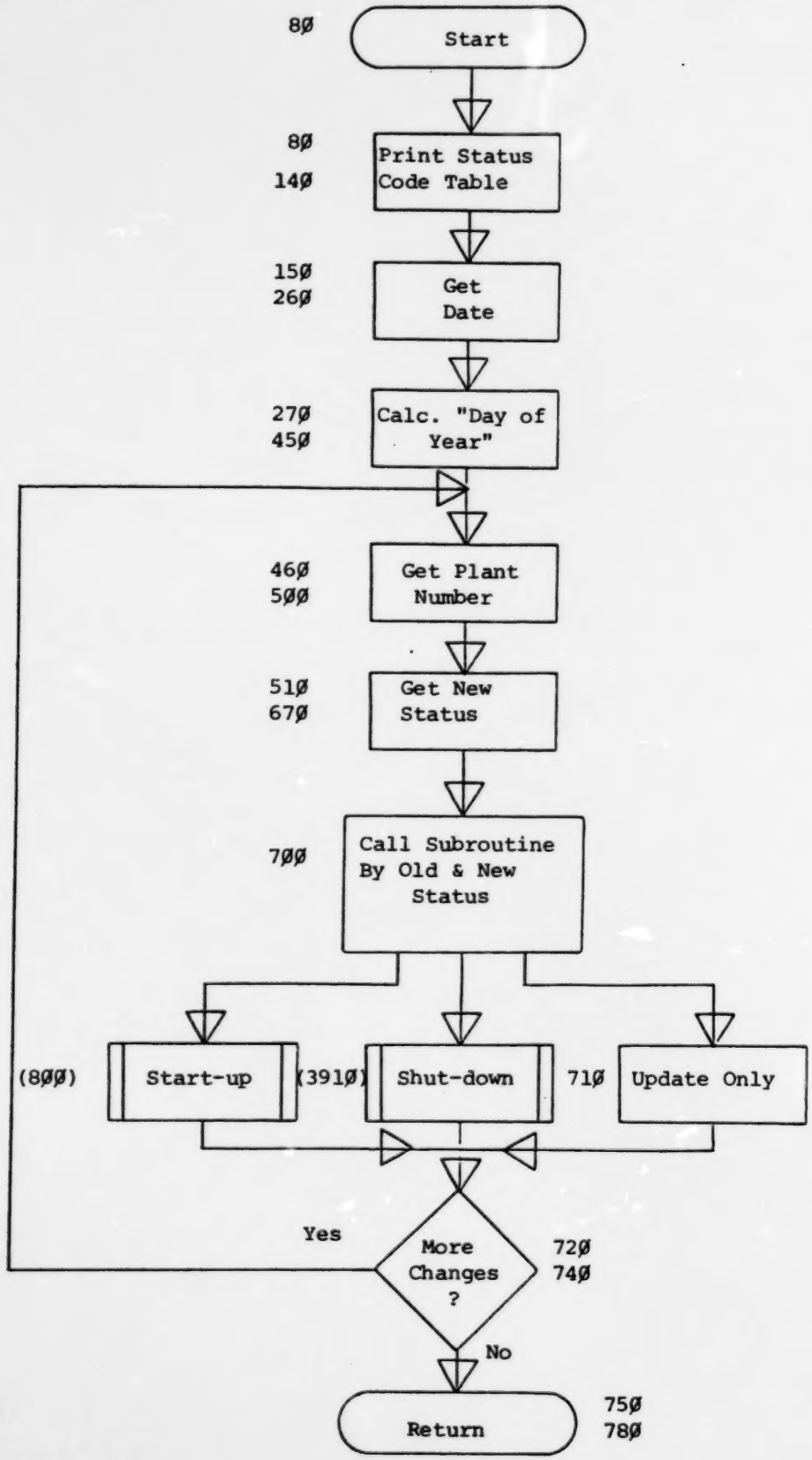
## File 4 - Operation Changes



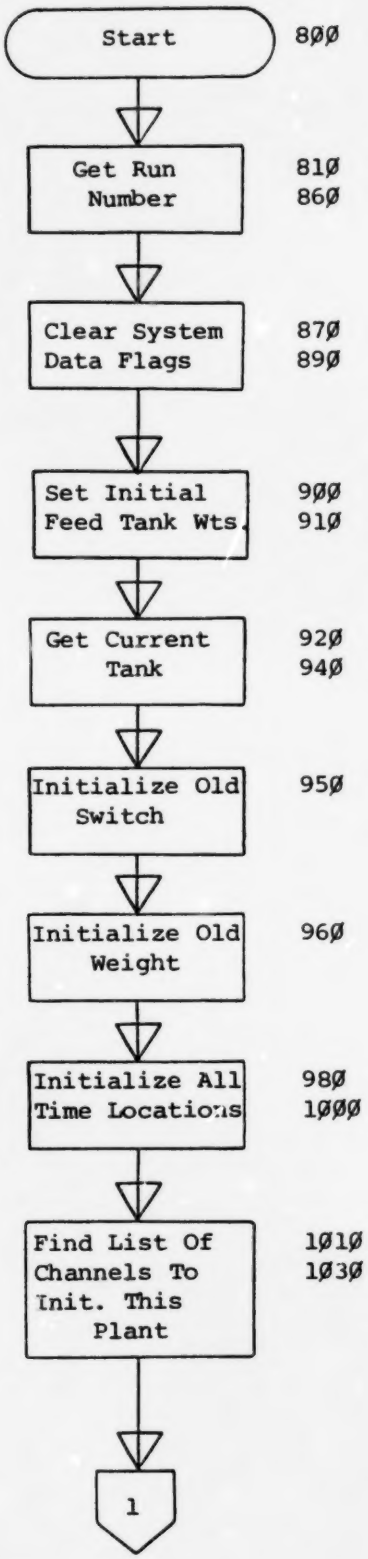
File 4 - Hourly Data & Report

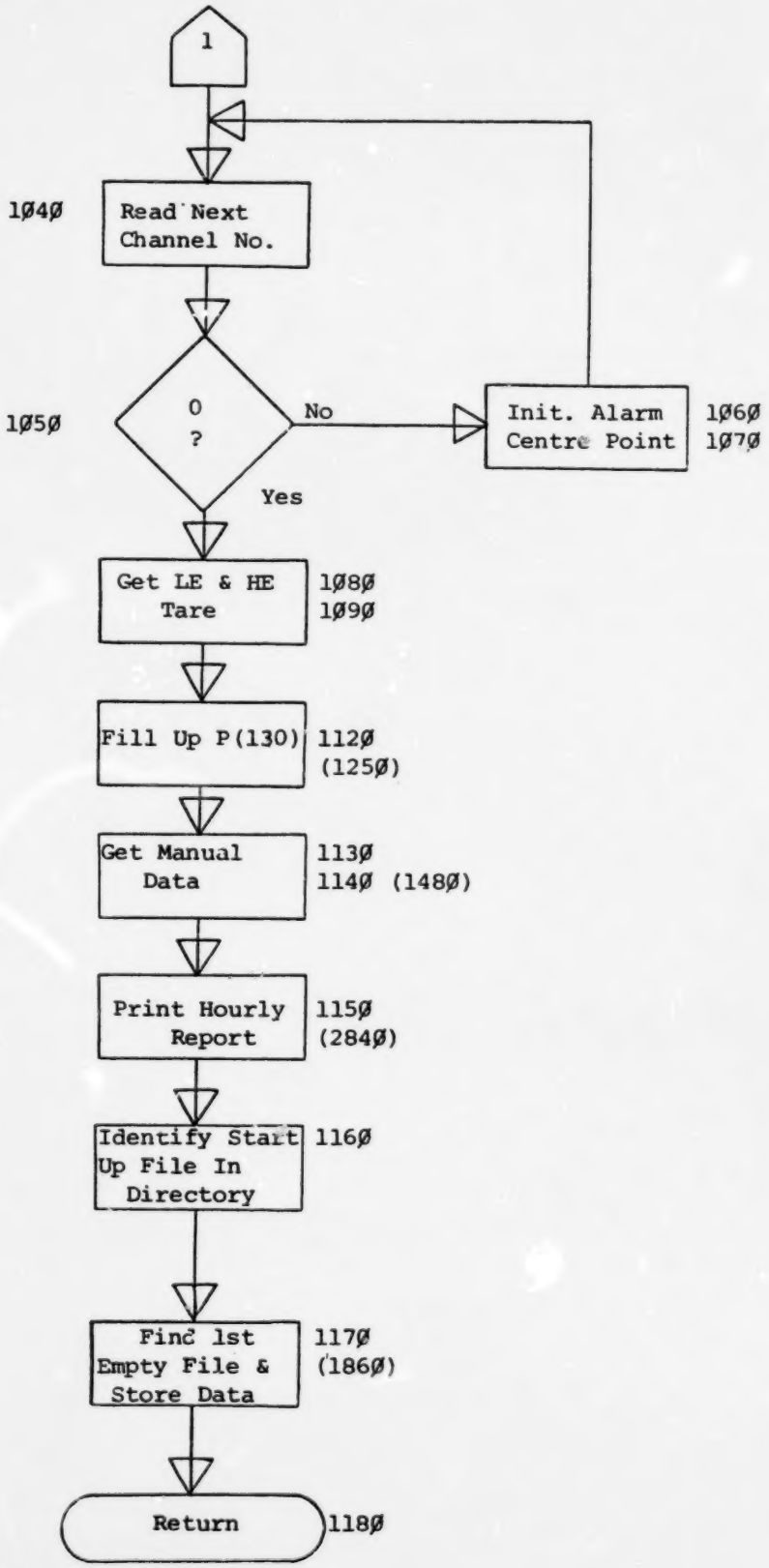


**File 4 - Plant Status Change**



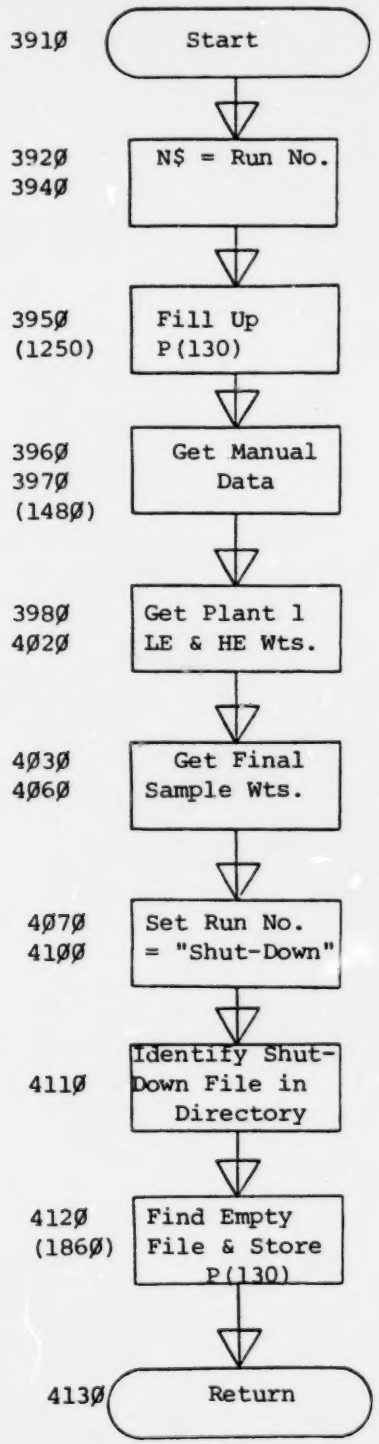
File 4 - Plant Status - Plant Start-up



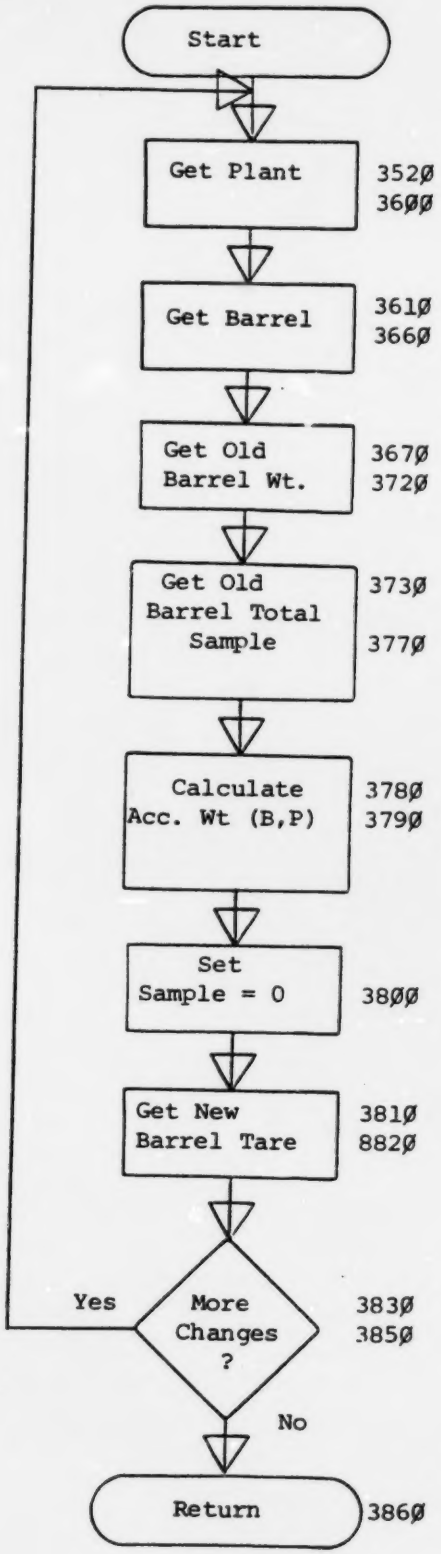

1
1Ø4Ø
Read Next Channel No.
1Ø5Ø
0 ?
No
Init. Alarm Centre Point
1Ø6Ø
1Ø7Ø
Yes
Get LE & HE Tare
1Ø8Ø
1Ø9Ø
Fill Up P(130)
112Ø
(125Ø)
Get Manual Data
113Ø
114Ø (148Ø)
Print Hourly Report
115Ø
(284Ø)
Identify Start Up File In Directory
116Ø
Find 1st Empty File & Store Data
117Ø
(186Ø)
Return
118Ø

File 4 - Plant Status - Plant Shut-Down

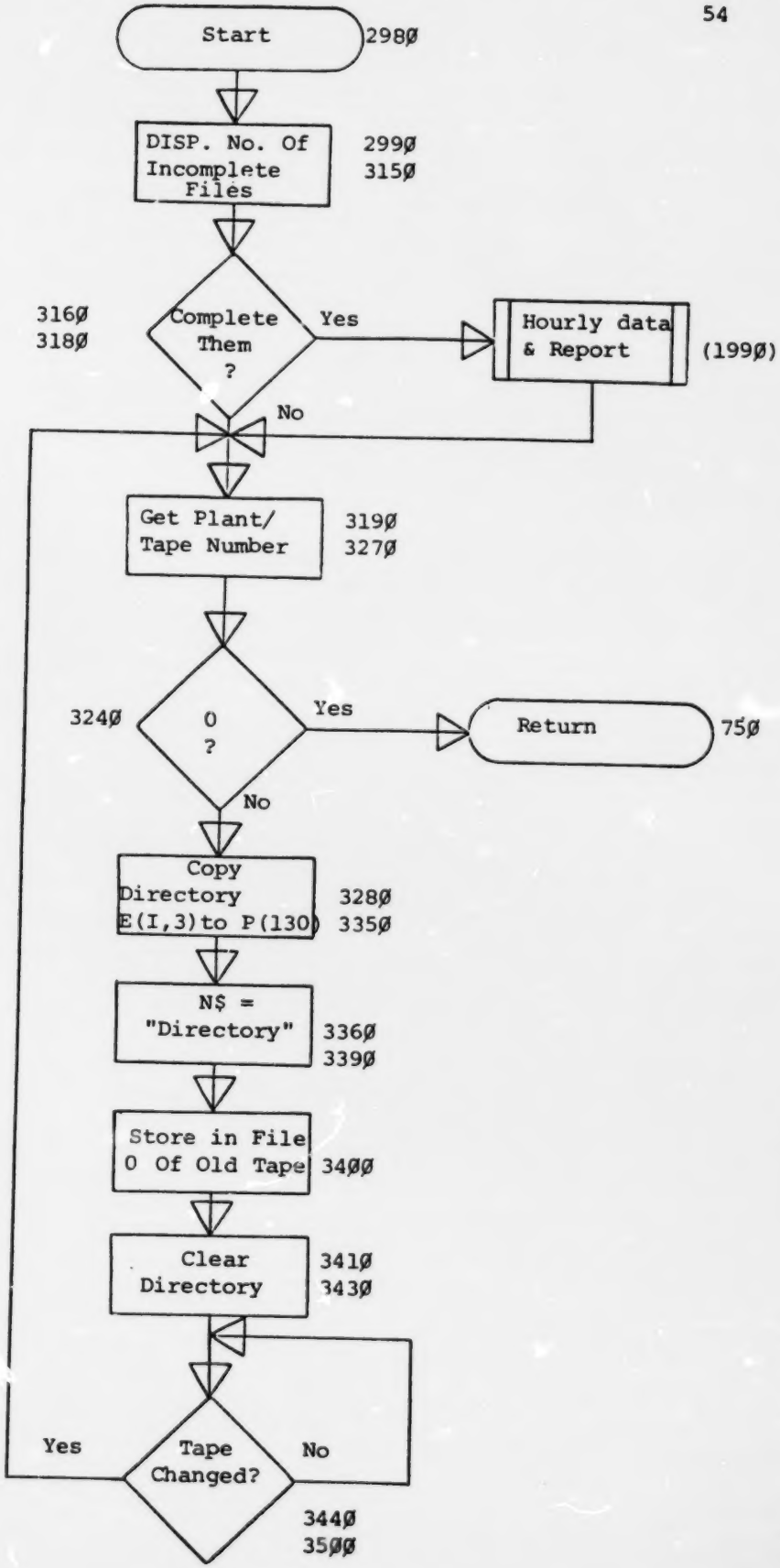




File 4 - Barrel Change



File 4 - Change Cassette



```
10 COM PS[130],H$[20],N$[20],T$[80],U$[80],V$[80]
20 DIM A$[20],B$[20],C$[20],D$[35],E$[3],F$[15],G$[80],M$[60],P$[30],S$[3]
30 DIM W$[80],X$[20],Y$[20],Z$[20],A$[200],C$[200,3]
40 DIM DI[6],EI[200,3],FI[2],GI[5,2],LI[6,6],TI[4,2],W[3,3],XI[6],YI[4]
50 REM---ENTRY POINT---
60 FIND 1
70 GOTO W OF 1990,80,3510,2980
80 REM---CHANGE PLANT STATUS---
90 PRINT "STATUS CODES:"
100 PRINT TAB(13);"0.  RUNNING"
110 PRINT TAB(13);"1.  ALARMS OFF"
120 PRINT TAB(13);"2.  SHUT DOWN"
130 PRINT
140 PRINT
150 DISP "ENTER TODAY'S DATE (DD,MM,YYYY)";
160 INPUT D,M,Y
170 M=M-1
180 IF Y>100 THEN 230
190 IF Y<50 THEN 220
200 Y=Y+1900
210 GOTO 230
220 Y=Y+2000
230 IF M >= 0 AND M<12 AND D<32 AND D>0 AND Y>1583 THEN 270
240 DISP "ERROR--TRY AGAIN"
250 WAIT 3000
260 GOTO 150
270 IF M>1 THEN 300
280 D2=365*Y+D+31*M+INT((Y-1)/4)-INT(3*(INT((Y-1)/100)+1)/4)
290 GOTO 310
300 D2=365*Y+D+31*M-INT(0.4*(M+1)+2.3)+INT(Y/4)-INT(0.75*(INT(Y/100)+1))
310 D1=365*Y+INT((Y-1)/4)-INT(3*(INT((Y-1)/100)+1)/4)+1
320 D3=D2-(INT(D2/7)*7)
330 D4=D2-D1+1
340 D5=365*Y+31*12-INT(0.4*(11+1)+2.3)+INT(Y/4)-INT(0.75*(INT(Y/100)+1))
350 C[187,3]=D5-D1+1
360 DISP "FOR "D$[D3*5+1,D3*5+5]"--"M$[M*5+1,M*5+5];D","Y
370 GOSUB 680
380 DISP "DAY OF THE YEAR = "D4
390 GOSUB 680
400 DISP "IS TODAY "D$[D3*5+1,D3*5+5];
410 INPUT A$
420 IF A$[1,1]="Y" THEN 440
430 GOTO 240
440 C[182,3]=D4
450 C[181,3]=Y
```

```
460 DISP "WHICH PLANT IS CHANGING STATUS";
470 INPUT P
480 IF P=1 OR P=2 THEN 510
490 GOSUB 3870
500 GOTO 440
510 DISP "ENTER NEW STATUS (0,1 OR 2)";
520 INPUT S
530 IF S >= 0 AND S<3 THEN 570
540 DISP "NOT A VALID STATUS CODE"
550 GOSUB 680
560 GOTO 510
570 PRINT "PLANT "P"GOING FROM---"P$[C[184+(P-1),3]*10+1,C[184+(P-1),3]*10+10
580 PRINT "--- TO ---"P$[S*10+1,S*10+10]
590 PRINT
600 PRINT
610 PRINT
620 DISP "IS THAT CORRECT (Y OR N)";
630 INPUT A$
640 IF A$[1,1]="Y" THEN 700
650 DISP "OK---TRY AGAIN"
660 GOSUB 680
670 GOTO 440
680 WAIT 2000
690 RETURN
700 GOSUB W[C[184+(P-1),3]+1,S+1] OF 3910,800
710 C[184+(P-1),3]=S
720 DISP "ARE THERE ANY MORE CHANGES";
730 INPUT A$
740 IF A$[1,1]="Y" THEN 440
750 REM---RETURN TO SCAN---
760 STORE  DATA 1,C
770 STORE  DATA 2,E
780 LINK 3,10,80
790 RETURN
800 REM---START UP---
810 DISP "ENTER NEW RUN NUMBER";
820 INPUT N$
830 IF P>1 THEN 860
840 Y$=N$
850 GOTO 870
860 Z$=N$
870 FOR I=1 TO 30
880 C[120+I+(P-1)*30,3]=0
890 NEXT I
900 C[121+(P-1)*30,3]=A[T[1,P]]
910 C[122+(P-1)*30,3]=A[T[2,P]]
920 S1=0
930 IF A[T[3,P]]>1.5 THEN 950
940 S1=1
950 C[124+(P-1)*30,3]=S1
960 C[125+(P-1)*30,3]=A[T[1+S1,P]]
```

```
970 REM---INIT FEED RATE ARRAY---
980 FOR I=0 TO 6 STEP 2
990 C[143+(P-1)*30+I+1,3]=C[183,3]
1000 NEXT I
1010 RESTORE 1240
1020 IF P>1 THEN 1040
1030 RESTORE 1220
1040 READ L
1050 IF L=0 THEN 1080
1060 C[L,1]=A[L]
1070 GOTO 1040
1080 FOR I=0 TO 1
1090 DISP "ENTER---"G$[I*10+1,I*10+10]"---TARE";
1100 INPUT C[128+I+(P-1)*30,3]
1110 NEXT I
1120 GOSUB 1250
1130 F6=1
1140 GOSUB 1480
1150 GOSUB 2840
1160 F7=25400
1170 GOSUB 1860
1180 RETURN
1190 REM---CHAN. NUMBERS FOR AUTO ALARM---
1200 REM---MUST END WITH "0"---
1210 REM---PLANT 1---
1220 DATA 14,16,17,18,19,20,56,57,58,0
1230 REM---PLANT 2---
1240 DATA 0
1250 REM---FILL UP P(130)----
1260 FOR I=1 TO 130
1270 P[I]=0
1280 NEXT I
1290 IF P=2 THEN 1340
1300 FOR I=1 TO 89
1310 P[I]=A[I]
1320 NEXT I
1330 IF P=1 THEN 1380
1340 FOR I=90 TO 189
1350 P[I-89]=A[I]
1360 NEXT I
1370 IF P=2 THEN 1410
1380 FOR I=190 TO 200
1390 P[I-100]=A[I]
1400 NEXT I
1410 FOR I=0 TO 15
1420 P[I+101]=C[121+I+(P-1)*30,3]
1430 NEXT I
1440 FOR I=0 TO 4
1450 P[I+126]=C[I+181,3]
1460 NEXT I
1470 RETURN
```

File 4

```
1480 REM---MANUAL DATA---
1490 DISP "ENTER YOUR IDENT.  NUMBER";
1500 INPUT P[117]
1510 FOR I=0 TO 3
1520 DISP "ENTER---"G$[I*10+1,I*10+10]"---OFF GAS";
1530 INPUT V
1540 P[118+I]=V*0.0283
1550 NEXT I
1560 N2=7
1570 IF F6 THEN 1620
1580 FOR I=0 TO 1
1590 DISP "ENTER---"G$[I*10+1,I*10+10]"---SAMPLES";
1600 INPUT P[110+I]
1610 NEXT I
1620 IF P=2 THEN 1690
1630 IF F6 THEN 1680
1640 FOR I=0 TO 1
1650 DISP "ENTER---"G$[I*10+1,I*10+10]"---WEIGHT";
1660 INPUT P[124+I]
1670 NEXT I
1680 N2=5
1690 FOR I=4 TO N2
1700 DISP "ENTER---"G$[I*10+1,I*10+10]"---WATT METER";
1710 INPUT P[118+I]
1720 NEXT I
1730 PRINT "ENTER COMMENTS (MAX.  3 LINES OF 80 CHARACTERS)"
1740 PRINT "IF NO COMMENT ENTER 'NONE'"
1750 PRINT
1760 PRINT
1770 PRINT
1780 INPUT T$
1790 IF T$#"NONE" THEN 1830
1800 U$=T$
1810 V$=T$
1820 GOTO 1850
1830 INPUT U$
1840 INPUT V$
1850 RETURN
1860 REM---FIND BLANK FILE AND STORE DATA---
1870 F=(P-1)*100+1
1880 IF E[F,3]=0 THEN 1910
1890 F=F+1
1900 GOTO 1880
1910 E[F,3]=F7
1920 STORE  DATA #5+(P-1),F-(P-1)*100
1930 IF F-(P-1)*100-1>50 THEN 1950
1940 RETURN
1950 PRINT "CHANGE TAPE CASSETTE"
1960 PRINT
1970 PRINT
1980 GOTO 1940
```

File 4

```
1990 REM---ENTER MANUAL DATA AND PRINT HOURLY REPORT---
2000 GOSUB 2530
2010 PRINT "TO EXIT ENTER 0,0"
2020 PRINT
2030 PRINT
2040 DISP "ENTER FILE NUMBER (PLANT,FILE)";
2050 INPUT P,F
2060 IF P=0 AND F=0 AND W=4 THEN 3190
2070 IF P=0 AND F=0 THEN 750
2080 IF P=2 OR P=1 THEN 2110
2090 GOSUB 3870
2100 GOTO 2040
2110 IF F>0 AND F<F[P] THEN 2150
2120 DISP "NO SUCH FILE"
2130 GOSUB 680
2140 GOTO 2040
2150 LOAD  DATA #5+(P-1),F
2160 IF E[F+(P-1)*100,3]-INT(E[F+(P-1)*100,3]/1000)*1000>366 THEN 2190
2170 F6=0
2180 GOSUB 1480
2190 GOSUB 2840
2200 DISP "DO YOU WANT TO CHANGE ANY NUMBERS";
2210 INPUT A$
2220 IF A$[1,1]#"Y" THEN 2370
2230 PRINT "ENTER SUBSCRIPT OF DATA TO BE CHANGED"
2240 PRINT "ENTER 0 TO EXIT"
2250 PRINT
2260 PRINT
2270 DISP "ENTER SUBSCRIPT";
2280 INPUT I
2290 IF I=0 THEN 2370
2300 IF I>0 AND I<131 THEN 2340
2310 DISP "NOT A VALID SUBSCRIPT "
2320 GOSUB 680
2330 GOTO 2270
2340 DISP "ENTER CORRECTED DATA";
2350 INPUT P[I]
2360 GOTO 2270
2370 DISP "WANT TO CHANGE THE COMMENTS";
2380 INPUT A$
2390 IF A$[1,1]#"Y" THEN 2490
2400 DISP "ENTER CORRECT RUN NUMBER";
2410 INPUT N$
2420 PRINT "RE-ENTER ALL THE COMMENTS WITH CORRECTIONS"
2430 PRINT "MAX.  3 LINES OF 80 CHARACTERS"
2440 PRINT
2450 PRINT
2460 INPUT T$
2470 INPUT U$
2480 INPUT V$
2490 STORE  DATA #5+(P-1),F
2500 IF E[F+(P-1)*100,3]-INT(E[F+(P-1)*100,3]/1000)*1000>366 THEN 2520
2510 E[F+(P-1)*100,3]=E[F+(P-1)*100,3]+366
2520 GOTO 1990
```

File 4

```
2530 REM---LIST DIRECTORY---
2540 PRINT TAB(12);"FILE DIRECTORY"
2550 FOR P=1 TO 2
2560 PRINT
2570 PRINT "************PLANT"P"*******************"
2580 PRINT "FILE  HR.     DAY                    STATUS"
2590 PRINT
2600 I=1
2610 IF E[I+(P-1)*100,3]=0 THEN 2660
2620 GOSUB 2690
2630 PRINT I;H;D,F$
2640 I=I+1
2650 GOTO 2610
2660 F[P]=I
2670 NEXT P
2680 RETURN
2690 IF E[I+(P-1)*100,3]>24000 THEN 2770
2700 H=INT(E[I+(P-1)*100,3]/1000)
2710 D=E[I+(P-1)*100,3]-H*1000
2720 F$="INCOMPLETE"
2730 IF D<367 THEN 2760
2740 D=D-366
2750 F$="COMPLETE"
2760 RETURN
2770 H=0
2780 D=0
2790 IF E[I+(P-1)*100,3] >= 26000 THEN 2820
2800 F$="START UP"
2810 RETURN
2820 F$="SHUT DOWN"
2830 RETURN
2840 REM---HOURLY REPORT---
2850 PRINT TAB(20);"HOURLY REPORT FOR  ";H$
2860 PRINT
2870 PRINT "SUBSCRIPT         DATA        SUBSCRIPT          DATA"
2880 FOR I=1 TO 65
2890 PRINT I,P[I],I+65,P[I+65]
2900 NEXT I
2910 PRINT N$
2920 PRINT T$
2930 PRINT U$
2940 PRINT V$
2950 PRINT
2960 PRINT
2970 RETURN
```

```
2980 REM---CHANGE CASSETTE ---
2990 REM---COUNT INCOMPLETE FILES---
3000 FOR P=1 TO 2
3010 I=1
3020 C=0
3030 IF E[I+(P-1)*100,3]=0 THEN 3070
3040 GOSUB 3120
3050 I=I+1
3060 GOTO 3030
3070 PRINT "PLANT"P"HAS"C"INCOMPLETE FILES"
3080 NEXT P
3090 PRINT
3100 PRINT
3110 GOTO 3160
3120 REM---COUNT---
3130 IF E[I,3]-INT(E[I,3]/1000)*1000>367 THEN 3150
3140 C=C+1
3150 RETURN
3160 DISP "DO YOU WANT TO COMPLETE THESE";
3170 INPUT A$
3180 IF A$[1,1]="Y" THEN 1990
3190 PRINT "ENTER 0 TO EXIT"
3200 PRINT
3210 PRINT
3220 DISP "WHICH TAPE--(PLANT 1 OR 2)";
3230 INPUT P
3240 IF P=0 THEN 750
3250 IF P=1 OR P=2 THEN 3280
3260 GOSUB 3870
3270 GOTO 3190
3280 FOR I=1 TO 130
3290 P[I]=0
3300 NEXT I
3310 I=1
3320 IF E[I+(P-1)*100,3]=0 THEN 3360
3330 P[I]=E[I+(P-1)*100,3]
3340 I=I+1
3350 GOTO 3320
3360 IF P=2 THEN 3390
3370 N$="DIRECTORY PLANT 1"
3380 GOTO 3400
3390 N$="DIRECTORY PLANT 2"
3400 STORE  DATA #5+(P-1),0
3410 FOR I=(P-1)*100+1 TO (P-1)*100+100
3420 E[I,3]=0
3430 NEXT I
3440 PRINT "DIRECTORY FOR PLANT"P"CLEARED, CHANGE CASSETTE"
3450 PRINT
3460 PRINT
3470 DISP "HAVE YOU CHANGED THE TAPE";
3480 INPUT A$
3490 IF A$[1,1]#"Y" THEN 3440
3500 GOTO 3190
```

```
3510 REM---BARREL CHANGE INFO. ---
3520 PRINT "TO IDENTIFY WHICH BARREL IS CHANGING"
3530 PRINT "ANSWER QUESTIONS ON DISPLAY"
3540 PRINT
3550 PRINT
3560 DISP "WHICH PLANT";
3570 INPUT P
3580 IF P=1 OR P=2 THEN 3610
3590 GOSUB 3870
3600 GOTO 3560
3610 DISP "WHICH BARREL (LE=1,HE=2)";
3620 INPUT B
3630 IF B=1 OR B=2 THEN 3670
3640 DISP "NO SUCH BARREL"
3650 GOSUB 680
3660 GOTO 3610
3670 PRINT "ENTER WT OF OLD BARREL JUST BEFORE CHANGE"
3680 PRINT "ON PLANT 1 THIS INCLUDES THE TARE"
3690 PRINT
3700 PRINT
3710 DISP "ENTER WT.  OF OLD BARREL";
3720 INPUT C[134+(B-1)+(P-1)*30,3]
3730 PRINT "ENTER TOTAL WEIGHT OF SAMPLES TAKEN FROM OLD BARREL"
3740 PRINT
3750 PRINT
3760 DISP "ENTER SAMPLE WT. ";
3770 INPUT C[130+(B-1)+(P-1)*30,3]
3780 W1=C[134+B-1+(P-1)*30,3]+C[128+B-1+(P-1)*30,3]*(P-2)+C[130+B-1+(P-1)*30,3
3790 C[132+B-1+(P-1)*30,3]=C[132+B+1+(P-1)*30,3]+W1
3800 C[130+(B-1)+(P-1)*30,3]=0
3810 DISP "ENTER TARE OF NEW BARREL";
3820 INPUT C[128+B-1+(P-1)*30,3]
3830 DISP "ARE THERE ANY MORE CHANGES";
3840 INPUT A$
3850 IF A$[1,1]="Y" THEN 3510
3860 GOTO 750
3870 REM---INVALID PLANT NO. ---
3880 DISP "NO SUCH PLANT"
3890 GOSUB 680
3900 RETURN
```

**File 4**

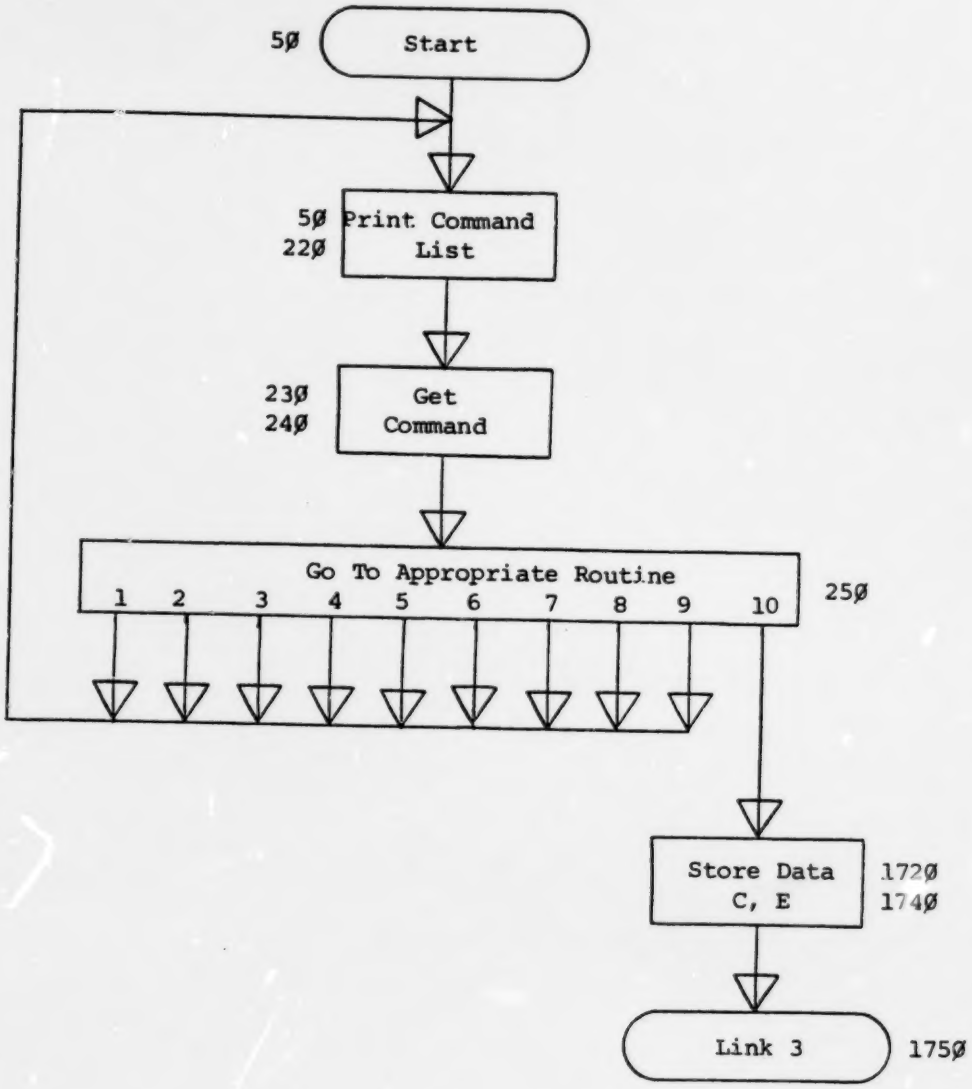
```
3910 REM---PLANT SHUT DOWN---
3920 N$=Y$
3930 IF P=1 THEN 3950
3940 N$=Z$
3950 GOSUB 1250
3960 F6=1
3970 GOSUB 1480
3980 IF P=2 THEN 4030
3990 FOR I=0 TO 1
4000 DISP "ENTER---"G$[I*10+1,I*10+10]"---FINAL WT. ";
4010 INPUT P[124+I]
4020 NEXT I
4030 FOR I=0 TO 1
4040 DISP "ENTER--"G$[I*10+1,I*10+10]"--FINAL SAMPLES";
4050 INPUT P[110+I]
4060 NEXT I
4070 IF P=2 THEN 4100
4080 Y$="SHUT DOWN"
4090 GOTO 4110
4100 Z$="SHUT DOWN"
4110 F7=26400
4120 GOSUB 1860
4130 RETURN
```

## 4.5 File 5 - Edit Options

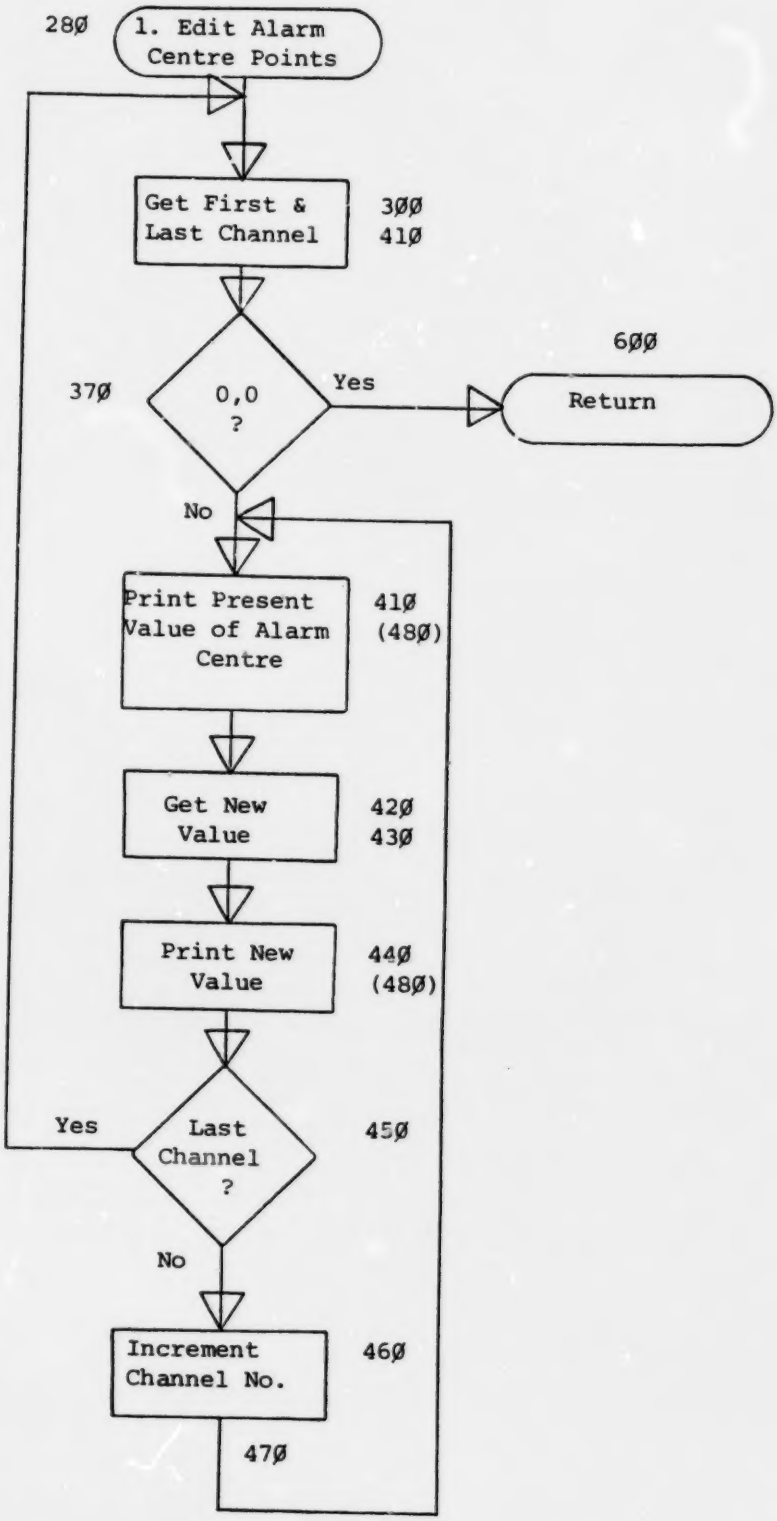
This program is linked into memory by the scanning program when the operator activates the interrupt switch and subsequently indicates a desire to "Edit Conversions, Alarm Limits, etc.". A further command list is printed which allows the operator to choose to edit alarm points or channel units and conversion factors, to set or clear print flags on channels to be monitored and to list the channels set to print, the current alarm data or the current conversion factors.

Each of the menu items calls a separate subroutine which interactively leads the operator through the necessary steps.

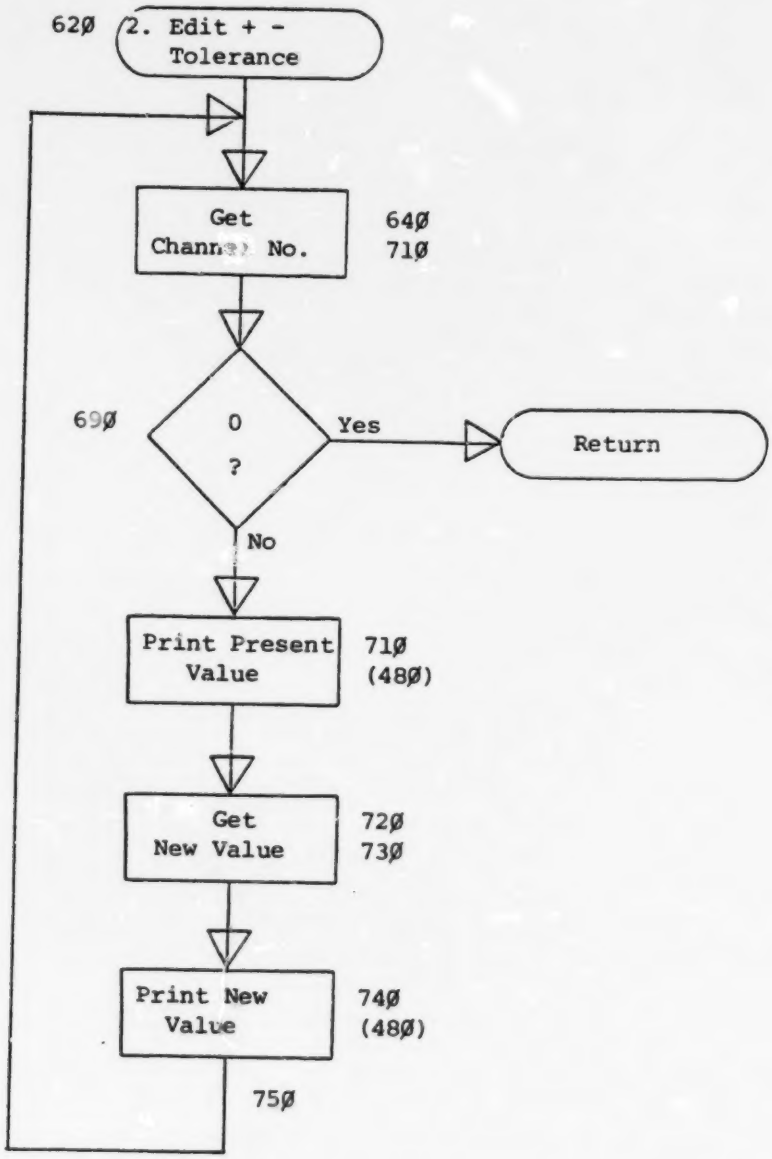
The changes made are not stored until the operator has made all the desired changes and wishes to return to the scan program.

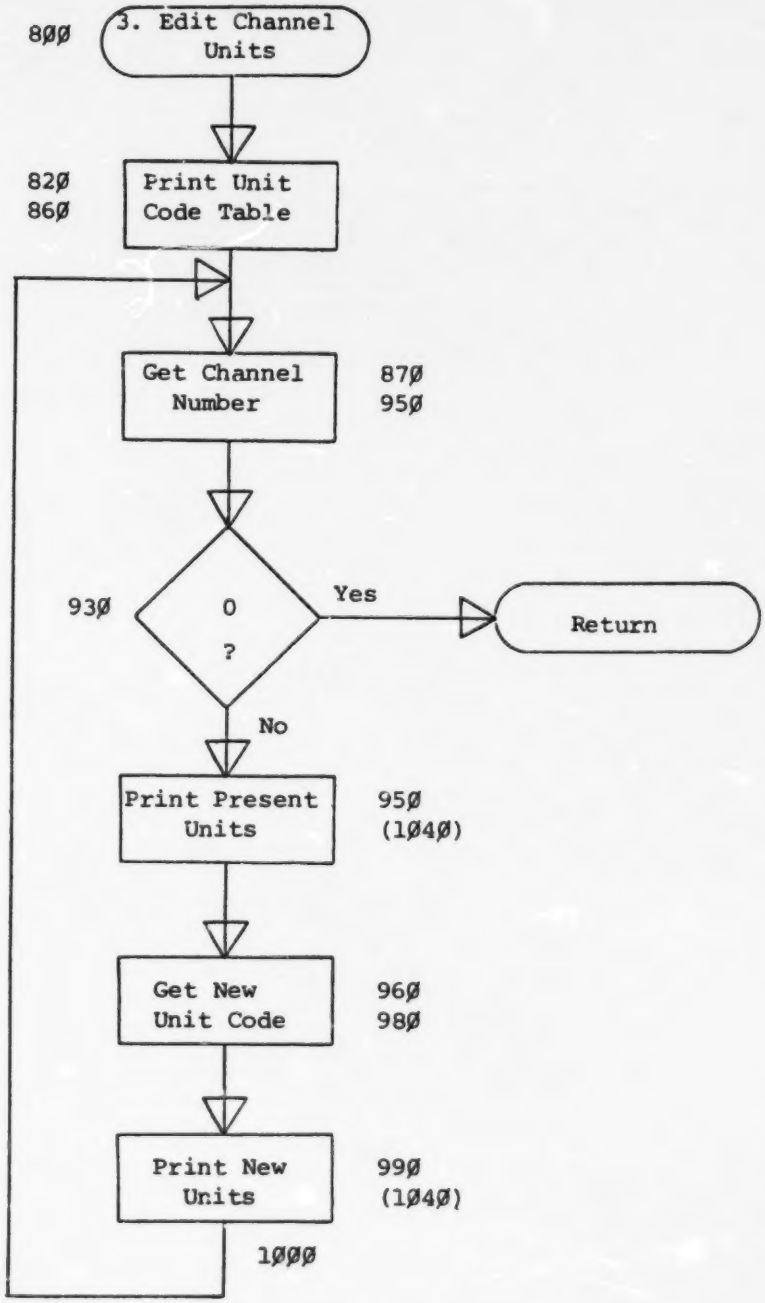

5Ø
Start
5Ø
22Ø
Print Command List
23Ø
24Ø
Get Command
Go To Appropriate Routine
1 2 3 4 5 6 7 8 9 10
25Ø
Store Data C, E
172Ø
174Ø
Link 3
175Ø

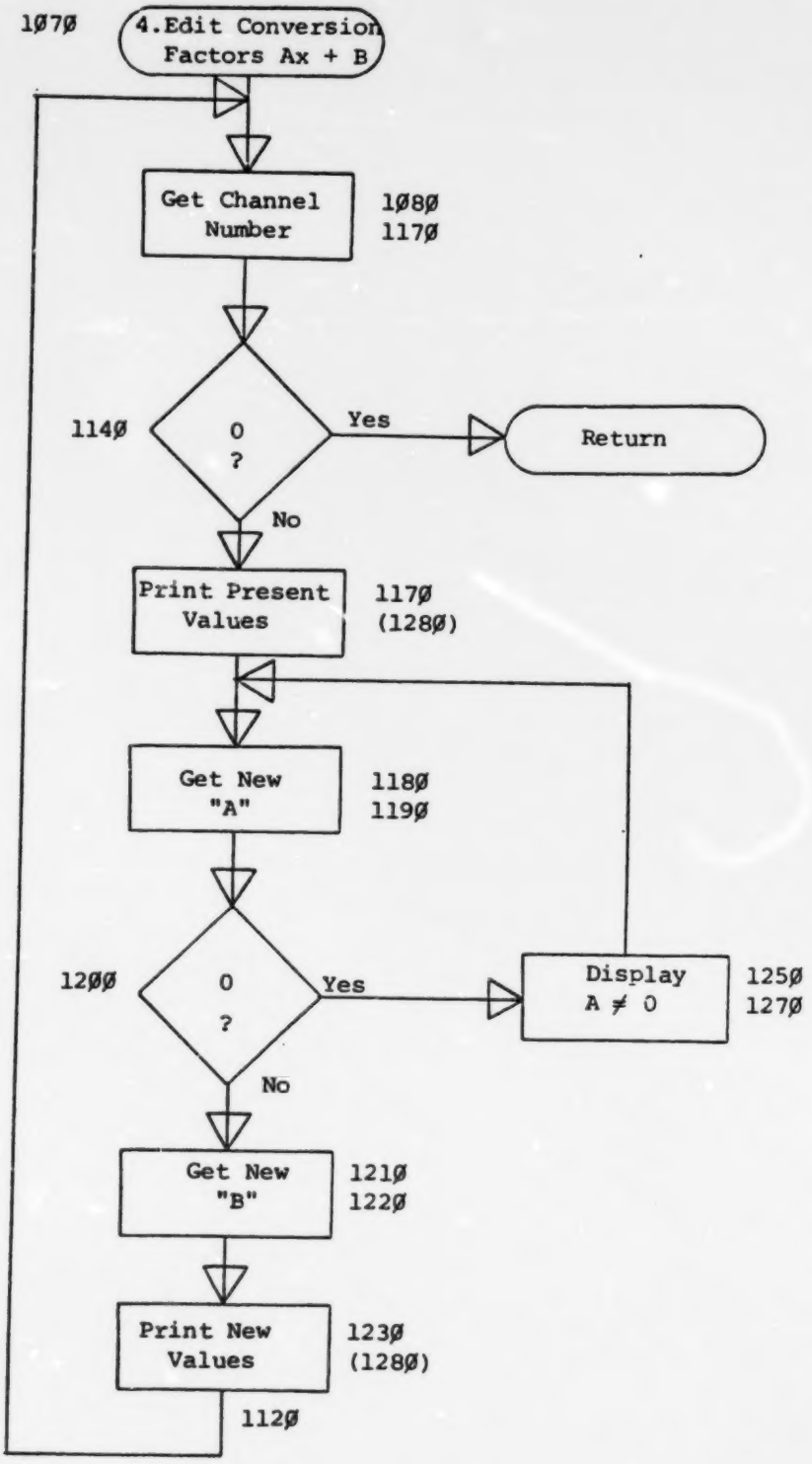
File 5 - Edit Options

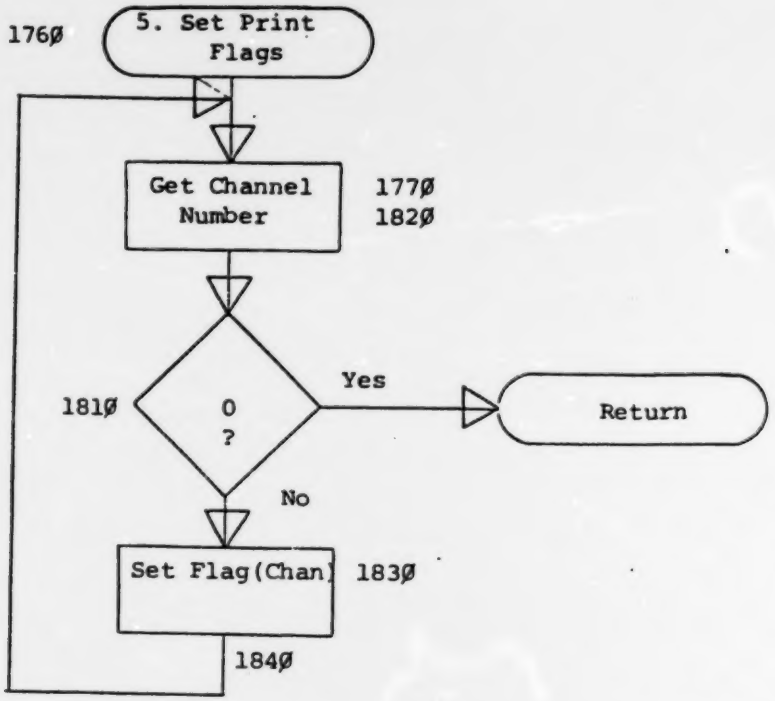


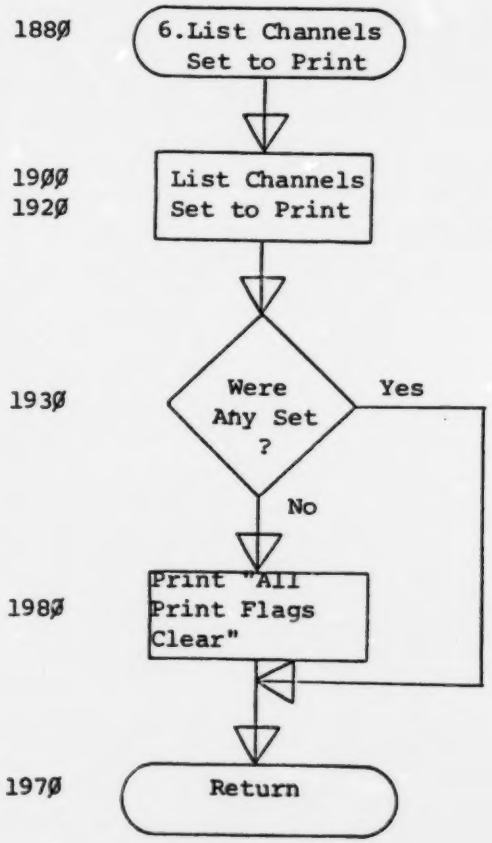
## File 5 - Edit Options

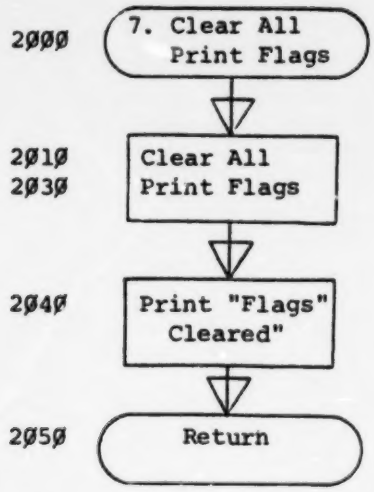


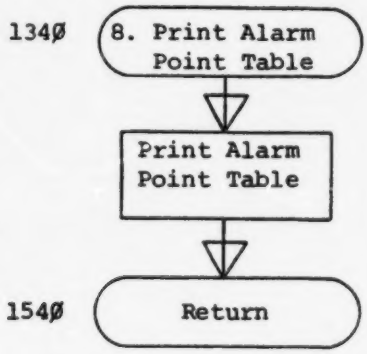

8ØØ
3. Edit Channel Units
82Ø
86Ø
Print Unit Code Table
Get Channel Number
87Ø
95Ø
93Ø
0 ?
Yes
Return
No
Print Present Units
95Ø
(1Ø4Ø)
Get New Unit Code
96Ø
98Ø
Print New Units
99Ø
(1Ø4Ø)
1ØØØ

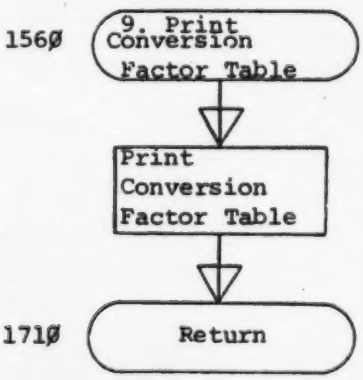

1070
4.Edit Conversion Factors Ax + B
Get Channel Number
1080
1170
1140
0 ?
Yes
Return
No
Print Present Values
1170
(1280)
Get New "A"
1180
1190
1200
0 ?
Yes
Display A ≠ 0
1250
1270
No
Get New "B"
1210
1220
Print New Values
1230
(1280)
1120


176Ø
5. Set Print Flags
Get Channel Number
177Ø
182Ø
181Ø
0 ?
Yes
Return
No
Set Flag(Chan)
183Ø
184Ø


188Ø
6.List Channels Set to Print
190Ø
192Ø
List Channels Set to Print
193Ø
Were Any Set ?
Yes
No
198Ø
Print "All Print Flags Clear"
197Ø
Return


2000
7. Clear All Print Flags
2010
2030
Clear All Print Flags
2040
Print "Flags" Cleared"
2050
Return


1340
8. Print Alarm Point Table
Print Alarm Point Table
1540
Return


1560
9. Print Conversion Factor Table
Print Conversion Factor Table
1710
Return

File 5

```
10 COM PS[130],H$[20],N$[20],T$[80],U$[80],V$[80]
20 DIM A$[20],B$[20],C$[20],D$[35],E$[3],F$[15],G$[80],M$[60],P$[30],S$[3]
30 DIM W$[80],X$[20],Y$[20],Z$[20],AS[200],CS[200,3]
40 DIM DI[6],EI[200,3],FI[2],GI[5,2],LI[6,6],TI[4,2],W[3,3],XI[6],YI[4]
50 REM---ENTRY POINT---
60 FIND 1
70 PRINT
80 PRINT
90 PRINT "COMMAND LIST:"
100 PRINT "1.   EDIT ALARM CENTER POINTS"
110 PRINT "2.   EDIT ALARM +- TOLERANCES"
120 PRINT "3.   EDIT CHANNEL UNITS"
130 PRINT "4.   EDIT CONVERSION FACTORS (AX+B)"
140 PRINT "5.   SET PRINT FLAGS"
150 PRINT "6.   LIST CHANNELS SET TO PRINT"
160 PRINT "7.   CLEAR ALL PRINT FLAGS"
170 PRINT "8.   PRINT ALARM DATA TABLE"
180 PRINT "9.   PRINT CONVERSION FACTOR TABLE"
190 PRINT "10.  UPDATE FILE AND EXIT"
200 PRINT
210 PRINT
220 PRINT
230 DISP "ENTER COMMAND NUMBER";
240 INPUT W
250 GOTO W OF 280,620,800,1070,1760,1880,2000,1340,1560,1720
260 GOTO 50
270 REM
```

File 5

```
280 REM---EDIT ALARM CENTER POINTS---
290 REM
300 PRINT "ENTER (FIRST CHANNEL),(LAST CHANNEL)"
310 PRINT "TO RETURN TO COMMAND MODE ENTER 0,0"
320 PRINT
330 PRINT
340 PRINT
350 DISP "CHANNEL NUMBERS (F,L)";
360 INPUT N,N1
370 IF N1=0 AND N=0 THEN 600
380 IF N<1 OR N1<1 THEN 540
390 IF N>200 OR N1>200 THEN 510
400 IF N1<N THEN 570
410 GOSUB 480
420 DISP "ENTER NEW CENTER VALUE CHAN. "N;
430 INPUT C[N,1]
440 GOSUB 480
450 IF N=N1 THEN 310
460 N=N+1
470 GOTO 410
480 PRINT "CHAN. "N"= "C[N,1]" +-"C[N,2];
490 PRINT W$[E[N,1]*5+1,E[N,1]*5+5]
500 RETURN
510 DISP "MAX. CHAN. NO. 200"
520 WAIT 3000
530 GOTO 350
540 DISP "CHAN. 0 NOT AVAILABLE"
550 WAIT 3000
560 GOTO 350
570 DISP "LAST MUST BE GREATER THAN FIRST"
580 WAIT 3000
590 GOTO 350
600 GOTO 50
610 REM
620 REM---EDIT TOLERANCE---
630 REM
640 PRINT "ENTER CHANNEL NUMBER"
650 PRINT "TO RETURN TO COMMAND MODE ENTER 0"
660 PRINT
670 DISP "ENTER CHANNEL NUMBER";
680 INPUT N
690 IF N=0 THEN 50
700 IF N<1 OR N>200 THEN 760
710 GOSUB 480
720 DISP "ENTER +- TOLERANCE";
730 INPUT C[N,2]
740 GOSUB 480
750 GOTO 670
760 DISP "1<CHAN. NO.<200"
770 WAIT 3000
780 GOTO 670
790 REM
```

File 5

```
800 REM---EDIT UNITS---
810 REM
820 PRINT "UNIT CODES"
830 FOR I=0 TO 11
840 PRINT I,W$[I*5+1,I*5+5]
850 NEXT I
860 PRINT
870 PRINT "ENTER CHANNEL NUMBER"
880 PRINT "TO RETURN TO COMMAND MODE ENTER 0"
890 PRINT
900 PRINT
910 DISP "ENTER CHANNEL NUMBER";
920 INPUT N
930 IF N=0 THEN 50
940 IF N<1 OR N>200 THEN 1010
950 GOSUB 1040
960 DISP "ENTER UNIT CODE";
970 INPUT U
980 E[N,1]=U
990 GOSUB 1040
1000 GOTO 910
1010 DISP "1<CHAN. NO. <200"
1020 WAIT 3000
1030 GOTO 910
1040 PRINT "CHAN.  ";N;"      ";W$[E[N,1]*5+1,E[N,1]*5+5]
1050 RETURN
1060 REM
1070 REM---EDIT CONVERSION FACTORS---
1080 REM
1090 PRINT "TO RETURN TO COMMAND MODE ENTER CHAN.  0"
1100 PRINT
1110 PRINT
1120 DISP "ENTER CHANNEL NUMBER";
1130 INPUT N
1140 IF N=0 THEN 50
1150 IF N<60 OR N>119 THEN 1300
1160 I=N-59
1170 GOSUB 1280
1180 DISP "ENTER NEW VALUE FOR 'A'";
1190 INPUT C[I,3]
1200 IF C[I,3]=0 THEN 1250
1210 DISP "ENTER NEW VALUE FOR 'B'";
1220 INPUT C[I+60,3]
1230 GOSUB 1280
1240 GOTO 1120
1250 DISP "'A' CANNOT BE 0"
1260 WAIT 3000
1270 GOTO 1180
1280 PRINT "CHAN. "N"A="C[I,3]"    B="C[I+60,3]
1290 RETURN
1300 DISP "CHAN. "N"HAS NO CONV.  FACTOR"
1310 WAIT 3000
1320 GOTO 1120
1330 REM
```

```
1340 REM---ALARM POINT TABLE---
1350 REM
1360 PRINT
1370 PRINT
1380 WRITE (15,1390)
1390 FORMAT 27"-","ALARM POINT TABLE",28"-"
1400 WRITE (15,1410)
1410 FORMAT 2" CHAN.      CENTER          +-           !"
1420 WRITE (15,1430)
1430 FORMAT 72"-"
1440 FOR I=1 TO 100
1450 GOSUB 1520
1460 I=I+100
1470 GOSUB 1520
1480 I=I-100
1490 PRINT
1500 NEXT I
1510 GOTO 50
1520 WRITE (15,1530)I,C[I,1],C[I,2],W$[E[I,1]*5+1,E[I,1]*5+5],"!";
1530 FORMAT F4.0,2F12.2,2X,F1.0
1540 RETURN
1550 REM
1560 REM---PRINT CONV.  FACTOR TABLE---
1570 REM
1580 PRINT
1590 PRINT
1600 WRITE (15,1610)
1610 FORMAT 27"-","CONV.  FACTOR TABLE",27"-"
1620 WRITE (15,1630)
1630 FORMAT 2" CHAN.          A               B          !"
1640 WRITE (15,1650)
1650 FORMAT 72"-"
1660 FOR I=1 TO 30
1670 N=I+59
1680 WRITE (15,1690)N,C[I,3],C[I+60,3],N+30,C[I+30,3],C[I+90,3]
1690 FORMAT F4.0,F12.2,F13.2,6X,"!",F4.0,F12.2,F13.2,6X,"!"
1700 NEXT I
1710 GOTO 50
1720 REM---UPDATE FILE AND EXIT---
1730 STORE  DATA 1,C
1740 STORE  DATA 2,E
1750 LINK 3,10,50
1760 REM---SET PRINT FLAGS---
1770 PRINT "ENTER CHAN.  NUMBER (0 TO EXIT)"
1780 PRINT
1790 DISP "ENTER CHAN.  NO.  ";
1800 INPUT N
1810 IF N=0 THEN 1890
1820 IF N>200 OR N<0 THEN 1850
1830 E[N,2]=1
1840 GOTO 1790
1850 DISP "NOT A VALID CHAN. "
1860 WAIT 3000
1870 GOTO 1790
```

```
1880 REM---LIST PRINTING CHANNELS---
1890 F2=1
1900 FOR I=1 TO 200
1910 IF E[I,2] THEN 1950
1920 NEXT I
1930 IF F2 THEN 1980
1940 GOTO 50
1950 PRINT "CHAN.  "I"SET TO PRINT ON EACH SCAN"
1960 F2=0
1970 GOTO 1920
1980 PRINT "ALL PRINT FLAGS CLEAR"
1990 GOTO 220
2000 REM---CLEAR ALL PRINT FLAGS---
2010 FOR I=1 TO 200
2020 E[I,2]=0
2030 NEXT I
2040 PRINT "FLAGS CLEARED"
2050 GOTO 220
```

## 4.6 File 6 - Program Fluke Channel Functions

This program is extra to the requirement but was written as a debugging aid for use in development of the rest of the system. In normal operation this program should not have to be used. However, in the case of major system changes in the future it may be found to be useful.

Due to the short and straightforward nature of the program, no flow chart is available.

The program provides for the maintenance of a data file (File 7) on the system tape, containing a Fluke program, for transferring this program to the Fluke data logger, for listing the program which is actually in the Fluke and for comparing the program in File 7 to the program in the Fluke data logger and noting any discrepancies.

The Fluke data logger retains its program memory even under power-down conditions so programming would normally be required only when channel functions had to be changed. Changing channel functions may require extensive modification of the rest of the software system.

File 6

```
10 DIM S[180,2],X$[100],A$[20],B$[20],E$[3]
20 FORMAT B
30 OUTPUT (E$,20)20
40 X$="SKIP 40 MV80 MV400MV4V   40V  T1   T2   T3   T4   NA   "
50 LOAD  DATA 7,S
60 FOR I=1 TO 180
70 S[I,2]=10
80 NEXT I
90 PRINT
100 PRINT
110 PRINT "   COMMAND LIST"
120 PRINT
130 PRINT "1.    PRINT CHANNEL FUNCTION TABLE"
140 PRINT "2.    EDIT CHANNEL FUNCTIONS"
150 PRINT "3.    STORE CHANNEL FUNCTION TABLE"
160 PRINT "4.    PROGRAM DATA LOGGER"
170 PRINT "5.    LIST DATALOGGER PROGRAM"
180 PRINT "6.    COMPARE FLUKE PROGRAM TO TABLE"
190 PRINT
200 PRINT
210 PRINT
220 DISP "INPUT COMMAND NO. ";
230 INPUT W
240 GOTO W OF 280,450,660,690,950,1260
250 DISP "NOT A VALID COMMAND"
260 WAIT 3000
270 GOTO 220
280 REM PRINT CHAN.  FUNCTION TABLE
290 WRITE (15,300)
300 FORMAT 26"-","CHAN.  FUNCTION TABLE",27"-"
310 WRITE (15,320)
320 FORMAT 4"CHAN.   FUNCTION   !"
330 WRITE (15,340)
340 FORMAT 72"-"
350 J=1
360 GOSUB 380
370 GOTO 90
380 FOR I=1 TO 180/4
390 PRINT I;S[I,J];X$[S[I,J]*5+1,S[I,J]*5+5];"!";
400 PRINT I+45;S[I+45,J];X$[S[I+45,J]*5+1,S[I+45,J]*5+5];"!";
410 PRINT I+90;S[I+90,J];X$[S[I+90,J]*5+1,S[I+90,J]*5+5];"!";
420 PRINT I+135;S[I+135,J];X$[S[I+135,J]*5+1,S[I+135,J]*5+5];"!"
430 NEXT I
440 RETURN
450 REM EDIT CHANNEL FUNCTIONS
460 DISP "INPUT CHAN.  NO.   (0 TO EXIT)";
470 INPUT C
480 IF C=0 THEN 90
490 IF C>180 OR C<0 THEN 600
500 GOSUB 570
```

File 6

```
510 DISP "INPUT NEW CHAN. FUNCTION";
520 INPUT A
530 IF A>9 OR A<0 THEN 630
540 S[C,1]=A
550 GOSUB 570
560 GOTO 460
570 I=S[C,1]*5+1
580 PRINT "CHAN. ";C;"=";S[C,1];"=";X$[I,I+4]
590 RETURN
600 DISP "CHAN. ";C;"NOT AVAILABLE"
610 WAIT 3000
620 GOTO 460
630 DISP "NOT A VALID FUNCTION"
640 WAIT 3000
650 GOTO 510
660 REM STORE CHAN. FUNCTION TABLE
670 STORE  DATA 7,S
680 GOTO 90
690 REM PROGRAM DATALOGGER
700 A$="E1"
710 GOSUB 910
720 A$="W4"
730 GOSUB 910
740 A$="S3"
750 GOSUB 910
760 ENTER (9,*)B$
770 IF B$[1,1]#"A" THEN 760
780 A$="S0"
790 GOSUB 910
800 ENTER (9,*)B$
810 IF B$[1,1]#E$[1,1] THEN 800
820 A$="C"
830 FOR I=1 TO 180
840 DISP "   PROGRAMMING DATALOGGER";
850 C=10000+I*10+S[I,1]
860 OUTPUT (B$,*)C
870 A$[2]=B$[3,7]
880 GOSUB 910
890 NEXT I
900 GOTO 90
910 OUTPUT (9,*)A$;
920 OUTPUT (9,20)13;
930 ENTER (9,*)B$
940 RETURN
950 REM LIST DATALOGGER PROGRAM
960 DISP "   INPUT FIRST CHANNEL";
970 INPUT C
980 A$="F"
990 GOSUB 1120
1000 GOSUB 910
```

```
1010 DISP "    INPUT LAST CHANNEL";
1020 INPUT C
1030 A$="L"
1040 GOSUB 1120
1050 GOSUB 910
1060 A$="P"
1070 GOSUB 910
1080 ENTER (9,*)B$
1090 IF B$[1,1]="@" THEN 1190
1100 IF B$[1,1]="C" THEN 1160
1110 GOTO 1080
1120 C=C+1000
1130 OUTPUT (B$,*)C
1140 A$[2]=B$[3,5]
1150 RETURN
1160 S[VAL(B$[2,4]),2]=VAL(B$[6,6])
1170 DISP "   READING FLUKE"
1180 GOTO 1080
1190 WRITE (15,1200)
1200 FORMAT 22"-","CHAN.  FUCTIONS NOW IN FLUKE",23"-"
1210 WRITE (15,320)
1220 WRITE (15,340)
1230 J=2
1240 GOSUB 380
1250 GOTO 90
1260 F=0
1270 FOR I=1 TO 180
1280 IF S[I,2]=10 THEN 1310
1290 F=1
1300 IF S[I,1]#S[I,2] THEN 1340
1310 NEXT I
1320 IF F=0 THEN 1370
1330 GOTO 90
1340 PRINT "ON CHAN. "; I; "  FLUKE READS  "; X$[S[I,2]*5+1,S[I,2]*5+5];
1350 PRINT "TABLE READS  "; X$[S[I,1]*5+1,S[I,1]*5+5]
1360 GOTO 1310
1370 PRINT "FLUKE HAS NOT BEEN READ YET"
1380 GOTO 1330
```

## 4.7 File 10 - Program for Copying Data Files

This is a short utility program of use in making back-up copies of the main system tape. Its use is described in the PHOIL System Operations Manual.

File 10

```
10 DIM CS[200,3],EI[200,3],SI[180,2]
20 LOAD  DATA #5,1,C
30 LOAD  DATA #5,2,E
40 LOAD  DATA #5,7,S
50 STORE  DATA 1,C
60 STORE  DATA 2,E
70 STORE  DATA 7,S
80 END
```

## 5. Considerations for Interfacing HP 9830A with 11205A Interface to Fluke 2240B Data Logger with -17 Option

### 5.1

Whenever an "S0" command is given to the data logger after a scan has begun, the data logger will respond with an @ CRLF followed by a DC4 (control T) CRLF. Both characters should be read before issuing any new commands. The response to the "S0" command, when the 2240B is not scanning is just @ CRLF.

### 5.2

The 2240B recognizes a CR as a command terminator. If an LF follows the command, it can cause the 2240B to stop issuing data. To insure the data logger continues to output data, the command should be followed only by a CR. This can be accomplished by using two output commands. The first issues the command with either a CR or LF; the second outputs the binary equivalent of a CR, such as

```
120 output (9,*) "S2" ;        This deletes CRLF after output
121 output (9,122) 13 ;
122 Format B
```

The first statement outputs scan control. The semicolon insures that no CRLF will follow. The second statement outputs the number 13 in binary format. This is the CR that causes the 2240B to process the first statement. Notice that it also must be followed by a semicolon. It also uses Format Statement 122 to output "13" binary.

"Q" command doesn't work on data loggers with software version 4 or less. This is the number that appears in the LSP on power up,

e.g. 2240B 4 - Main frame software version

5.3

The HP interface doesn't look at either Data Terminal Ready (pin 20) or Request to Send (pin 4). The Fluke data logger looks at pin 8 for a high level. The HP unit doesn't provide a high level on pin 8. Therefore pin 8 should be tied to pin 6 on the HP end of the cable or pin 20 may be tied to pin 8 on the Fluke end of the cable. It is not necessary to open the wire in the cable.

To ensure proper operation of the program list feature ( P command), the hardware echo on the HP interface must be disabled. This is done by cutting the land between pin 10 and 11 on U7 and shorting pin 11 to the land. Also, the 200 ms. delay after control characters was eliminated by lifting one end of R11. All numbers refer to revision B PCB.

5.4

When the HP 9830 wants to receive serial data from the Fluke data logger, an "ENTER" instruction is executed. Once the 9830 encounters this instruction in a program, it will wait for a line of data from the serial interface. This line must consist of one or more characters followed by a carriage return/line feed. If a line of data does not arrive at the serial interface, the 9830 will wait until one arrives or the operator restarts the program. There is no way in which the calculator can recover from this situation without operator intervention.

The 9830, therefore, must either know in advance how many lines of data will be sent by the Fluke or must be able to recognize the last line in a series in order to avoid executing an "ENTER" instruction for which there is no corresponding line of data.

Due to awkward software design in the Fluke, the number of lines of data or response varies not only from command to command but one command may give as many as three different sequences of response depending on the state of the Fluke when the command was received. For instance, the "S0" or end of scan command will respond as shown below.

| ANALOG DATA BEING SENT | SCAN FINISHED | TIME, ETC. BEING SENT |
|---|---|---|
| @(CRLF) | @(CRLF) | @(CRLF) |
| (CRLF) | | (Control) T (CRLF) |
| (Control) T (CRLF) | | |

Generally, the 9830 knows what state the Fluke is in before sending a command. However, when a power failure occurs, the 9830 must be able to regain control of the Fluke from an unkown state. This explains why the program initiates some seemingly unnecessary scans during initialization.

5.5

Another difficulty in the Fluke software makes it necessary to send a "W4" command before each scan. This is because the random access command, which, according to the Fluke manuals, should not affect memory, causes the Fluke to stop sending scan data.

Commands are sent to the Fluke using an "OUTPUT" instruction. Normally, the 9830 sends a CRLF at the end of each instruction. The Fluke, unfortunately, is programmed to receive only a CR character as a line terminator. To solve this problem, the CRLF is suppressed and a CR is sent separately after each command.

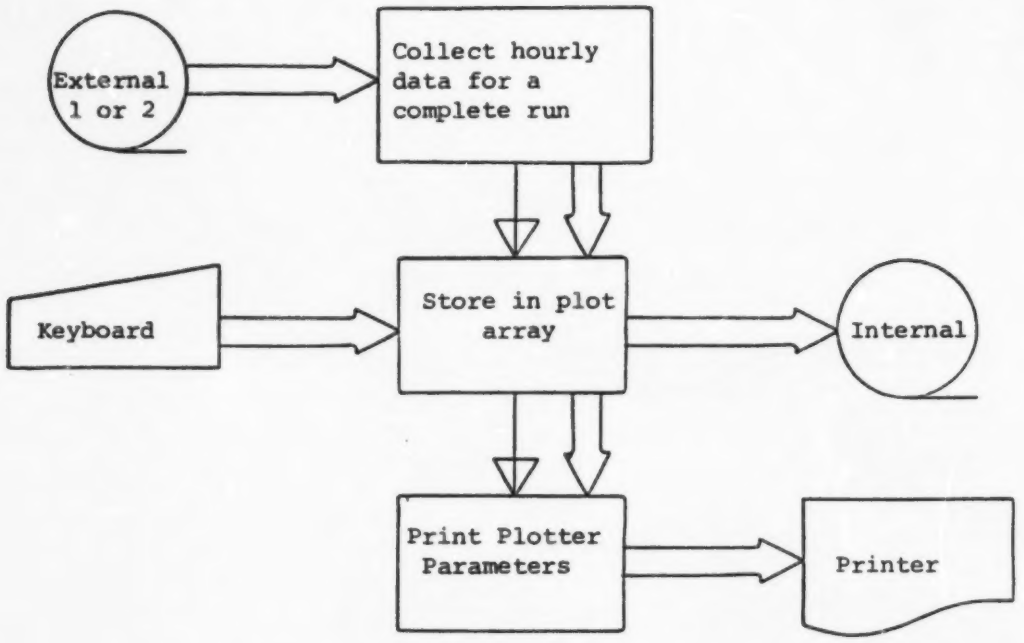
# 6. Data Extraction and Plotting Programs

## 6.1 Introduction

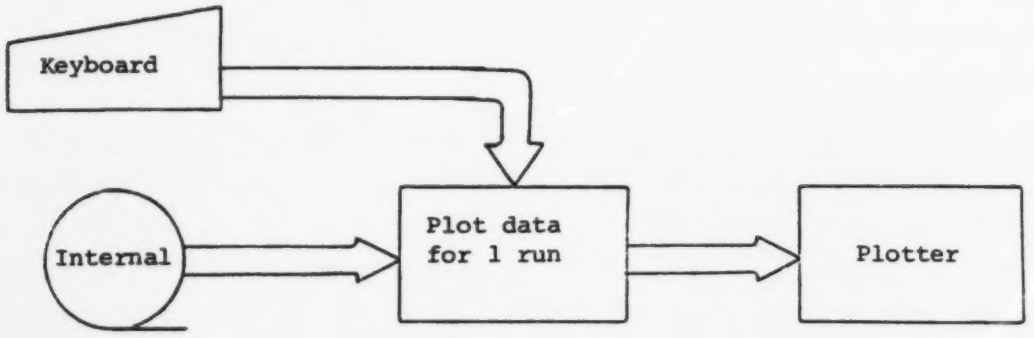
Two programs are provided on a separate cassette tape which allow retrieval of data from the data tapes and plotting of up to 256 points. These programs require a 9830A calculator with either the 11271B Plotter Control ROM or the Option 271 Plotter Control ROM and, of course, a graphics plotter.

Up to four variables may be extracted at one time from up to 256 files stored on a number of cassettes. The user is required to write a program for insertion into the data extraction program. This user program may combine the four variables in any appropriate way into a 256 x 2 array which the plotting program will then plot.

Data Retrieval




Plotting



## 6.3 Data Extraction Program

This program allows the operator to extract up to four variables from up to 256 files stored on several cassettes during a run. The program will decide whether to extract data from every 1,2 or 3 files depending on the length of the run.

This array is pre-formatted to conform to the plotting program.

The program will read the appropriate files between a "Start-up" and a "Shut-down" file and each time it runs out of files on a tape, it will request that the operator insert the next tape.

When the "Shut-down" file is encountered, it will execute the user program which must be entered into the space provided before executing the program.

If a location in the array is not written into, due to a missing file, an "Invalid data" flag will not be cleared and that location will be ignored on all subsequent operations.

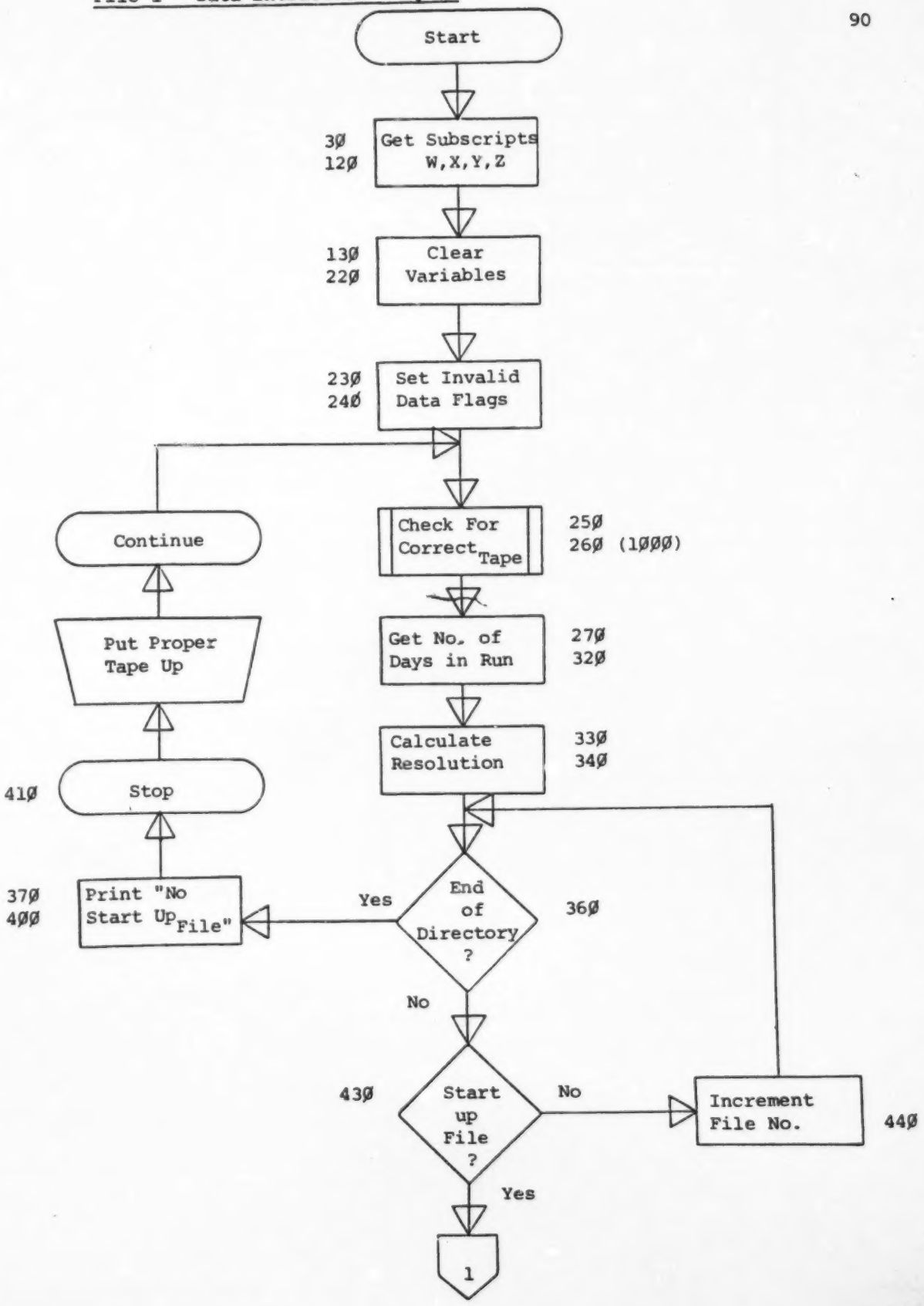
The user program must combine the four variables which have been extracted from the data files into one array A(256,2) using any formula. A line must be provided in this loop to ignore invalid data locations because an overflow may occur. The "REM" statements in the source listing give an example of how this should be done. This array, A(256,2), can then be stored for use by the plotting program.

The program will then print out some information on the array which will help in setting up the axes of the graph.

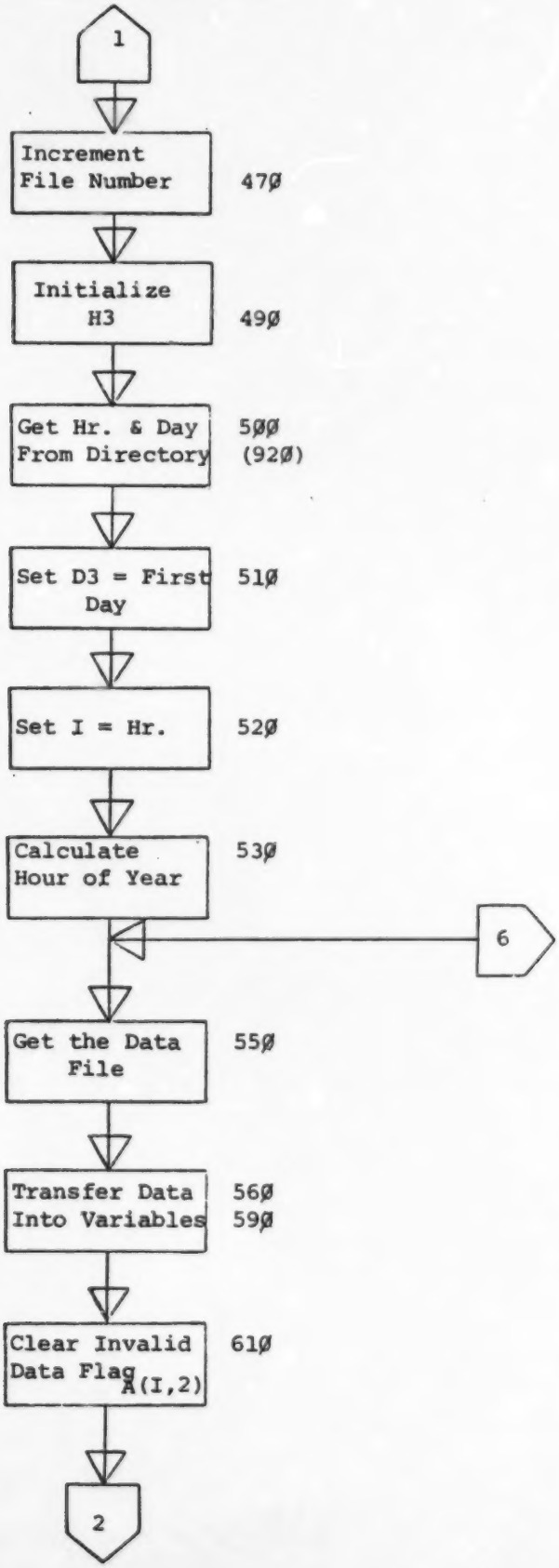
The four variables are stored in four arrays W(256), X(256), Y(256) and Z(256).

Detailed information on the user program requirements are to be found at Line 1430 in the program.

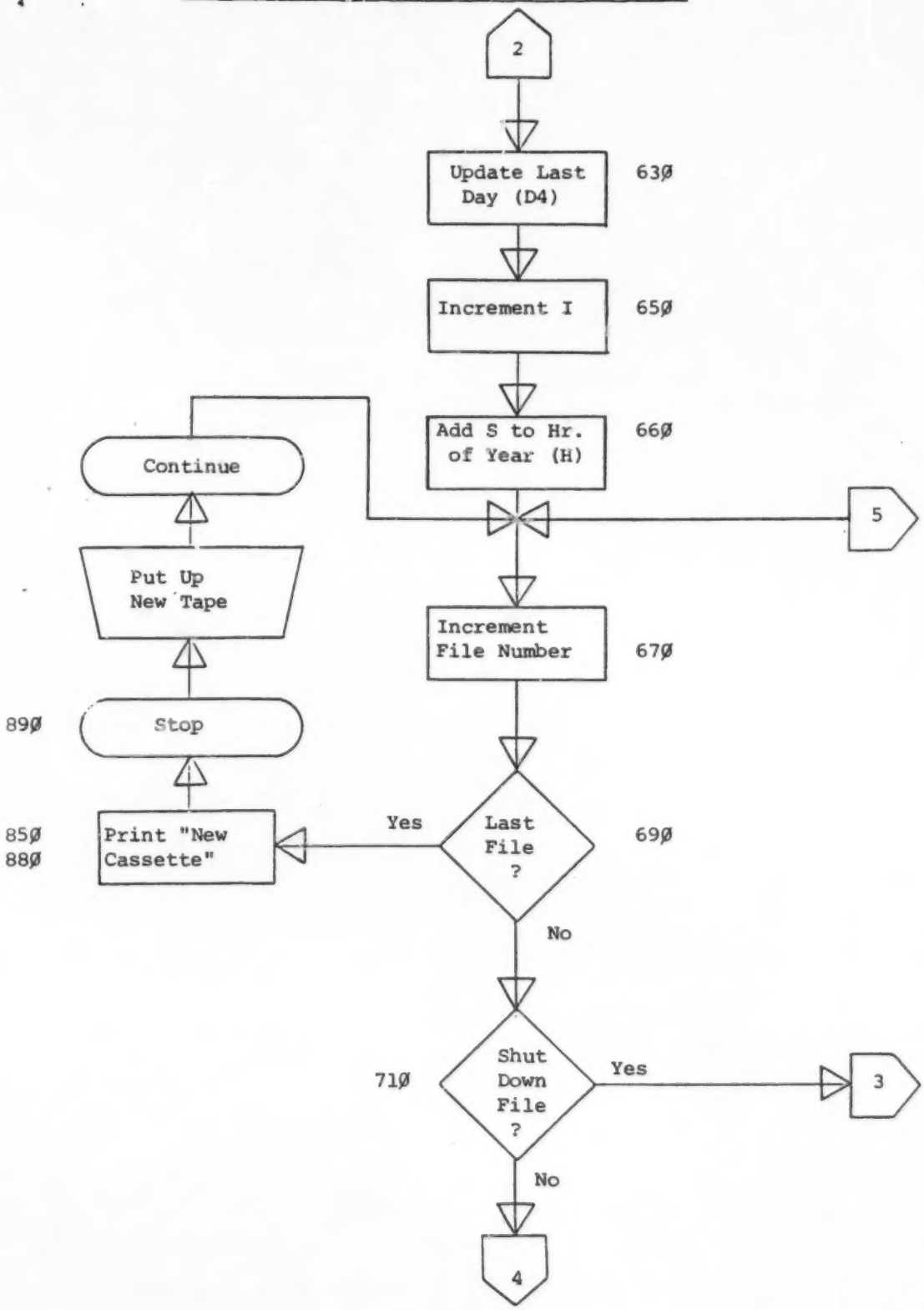
File 1 - Data Extraction Program



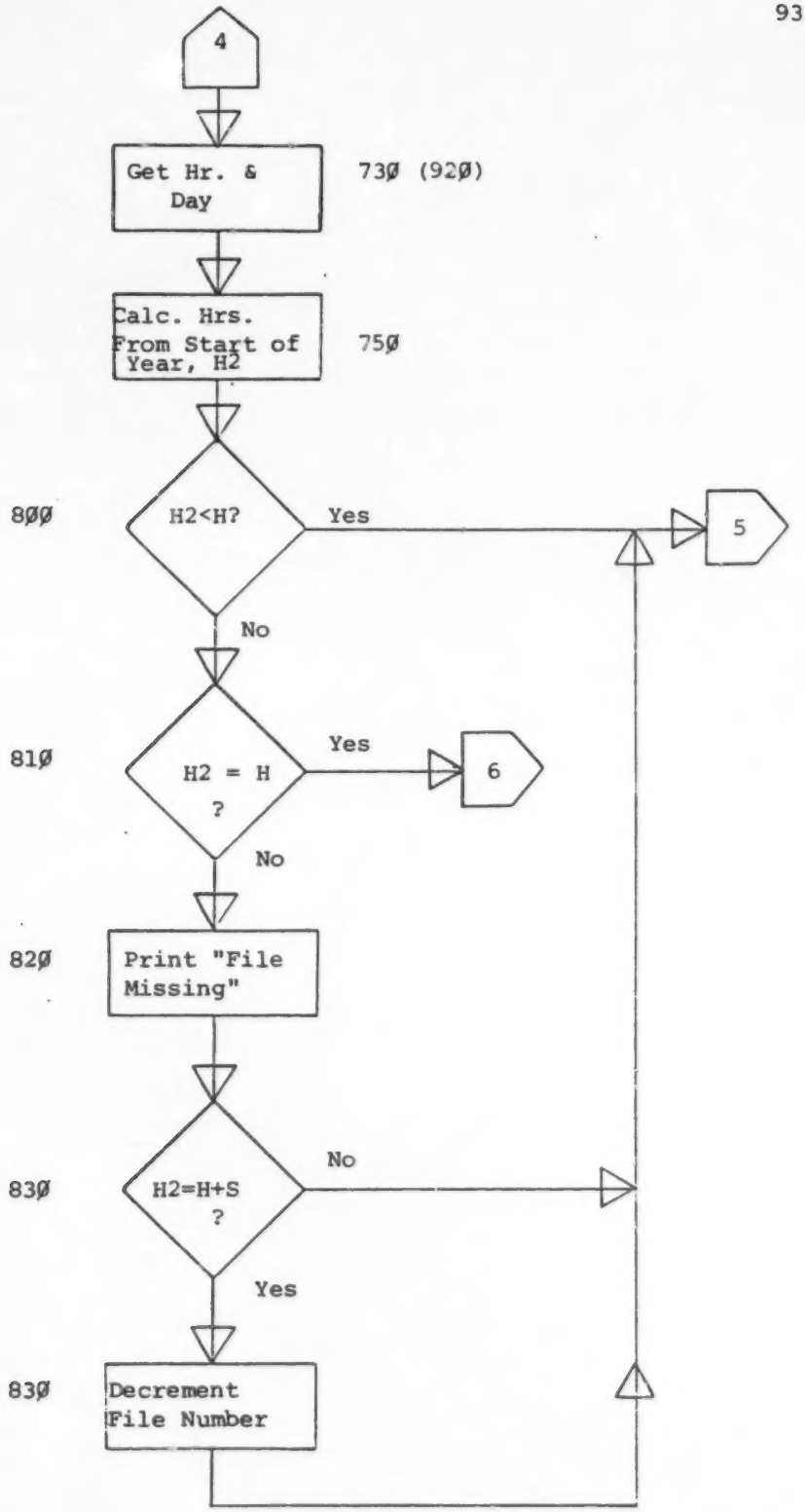
File 1 - Data Extraction Program (cont'd)

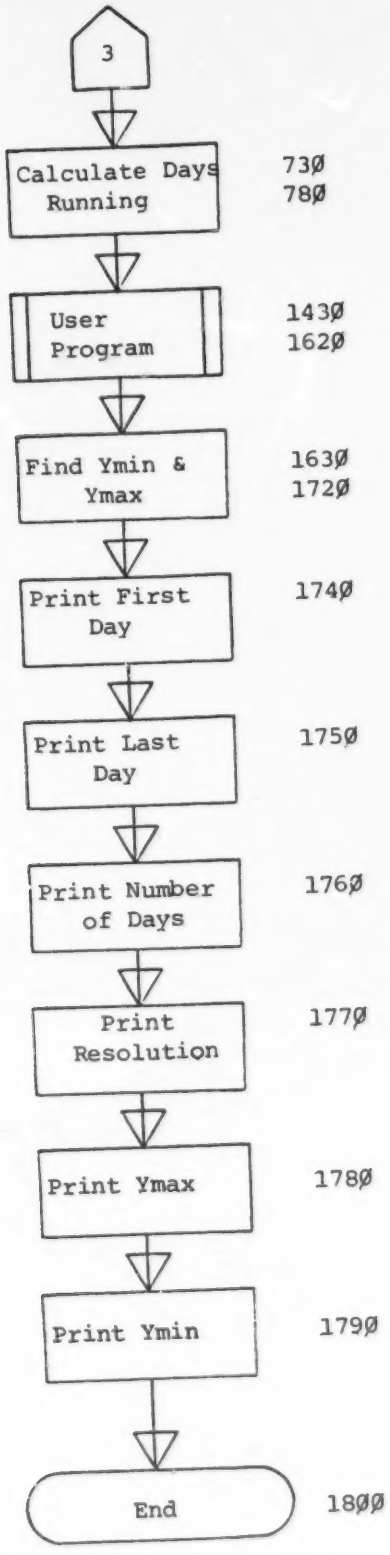


File 1 - Data Extraction Program (Cont'd)



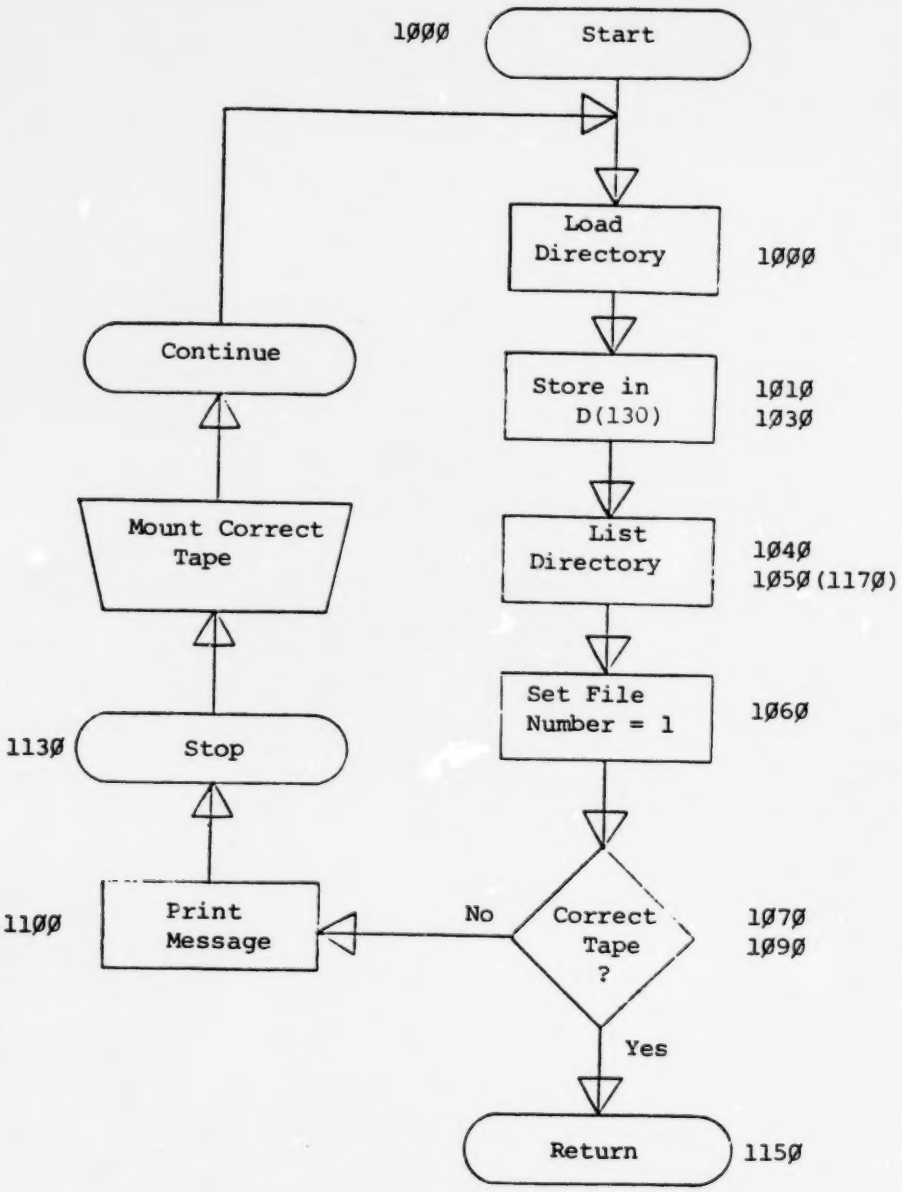
File 1 - Data Extraction Program (Cont'd)




3
Calculate Days Running
730
780
User Program
1430
1620
Find Ymin & Ymax
1630
1720
Print First Day
1740
Print Last Day
1750
Print Number of Days
1760
Print Resolution
1770
Print Ymax
1780
Print Ymin
1790
End
1800

File 1 - Data Retrieval Program - Check for Correct Tape Subroutine





File 1

```
10 COM P$[130],H$[20],N$[20],T$[80],U$[80],V$[80]
20 DIM D[130],A[256,2],W$[256],X$[256],Y$[256],Z$[256],F$[20]
30 REM---FOUR VARIABLES CAN BE EXTRACTED FROM P(130)---
40 REM---FOR USE IN CALCULATING VALUES TO BE PLOTTED---
50 DISP "ENTER SUBSCRIPT FOR 'W'";
60 INPUT W
70 DISP "ENTER SUBSCRIPT FOR 'X'";
80 INPUT X
90 DISP "ENTER SUBSCRIPT FOR 'Y'";
100 INPUT Y
110 DISP "ENTER SUBSCRIPT FOR 'Z'";
120 INPUT Z
130 REM---INIT. ARRAYS---
140 PRINT "STAND BY"
150 PRINT
160 PRINT
170 FOR I=1 TO 256
180 W[I]=0
190 X[I]=0
200 Y[I]=0
210 Z[I]=0
220 A[I,1]=0
230 A[I,2]=1
240 NEXT I
250 REM---GET DIRECTORY AND LIST IT---
260 GOSUB 1000
270 REM---THIS IS USED TO KEEP THE ARRAY SMALLER THAN 256---
280 DISP "ENTER NUMBER OF DAYS IN RUN";
290 INPUT D
300 IF D>0 AND D<23 THEN 340
310 DISP "MUST BE BETWEEN 1 AND 22 DAYS"
320 GOSUB 970
330 REM---DATA TAKEN EVERY 1,2 OR 3 HOURS DEPENDING ON LENGTH OF RUN---
340 S=INT(D*24/256)+1
350 REM---FIND START-UP FILE---
360 IF D[F]#0 THEN 430
370 PRINT "NO START UP FILE"
380 PRINT "PUT IN CORRECT CASSETTE AND 'CONT'"
390 PRINT
400 PRINT
410 STOP
420 GOTO 260
430 IF D[F]=25400 THEN 470
440 F=F+1
450 GOTO 360
460 REM---START WITH FILE FOLLOWING START-UP---
470 F=F+1
480 REM---GET INITIAL CONDITIONS---
490 H3=0
500 GOSUB 920
510 D3=D1
520 I=H1
530 H=H1+D1*24
```

```
540 REM---GET A FILE AND EXTRACT DATA---
550 LOAD  DATA #5,F
560 W[I]=P[W]
570 X[I]=P[X]
580 Y[I]=P[Y]
590 Z[I]=P[Z]
600 REM---SET VALID DATA---
610 A[I,2]=0
620 REM---UPDATE LAST DAY---
630 D4=D1
640 REM---INC.  POINTERS---
650 I=I+1
660 H=H+S
670 F=F+1
680 REM---END OF TAPE ? ---
690 IF D[F]=0 THEN 850
700 REM---END OF RUN ? ---
710 IF D[F]=26400 THEN 1410
720 REM---GET HR AND DAY---
730 GOSUB 920
740 REM---CALCULATE HOURS FROM BEGINNING OF YEAR---
750 H2=D1*24+H1+H3*(D1<D3)
760 REM---NEW YEAR---
770 IF D1<D3 THEN 800
780 H3=H2
790 REM---IS THIS THE RIGHT FILE---
800 IF H2<H THEN 670
810 IF H2=H THEN 550
820 PRINT "FILE MISSING AT HR="H-D1*24
830 F=F-(H2=H+S)
840 GOTO 650
850 PRINT "PUT NEXT CASSETTE IN EXTERNAL DRIVE"
860 PRINT "THEN HIT 'CONT' BUTTON"
870 PRINT
880 PRINT
890 STOP
900 GOSUB 1000
910 GOTO 670
920 REM---GET HR AND DAY FROM DIRECTORY ENTRY---
930 H1=INT(D[F]/1000)
940 D1=D[F]-H1*1000-366*(D[F]-H1*1000>366)
950 RETURN
960 REM---DISPLAY SUBROUTINE---
970 WAIT 3000
980 RETURN
```

```
990 REM---GET DIRECTORY,SAVE IT AND LIST IT---
1000 LOAD  DATA #5,0
1010 FOR K=1 TO 130
1020 D[K]=P[K]
1030 NEXT K
1040 PRINT "************";N$;"***************"
1050 GOSUB 1170
1060 F=1
1070 DISP "IS THAT THE CORRECT TAPE";
1080 INPUT N$
1090 IF N$[1,1]="Y" THEN 1150
1100 PRINT "PUT IN CORRECT CASSETTE AND 'CONT'"
1110 PRINT
1120 PRINT
1130 STOP
1140 GOTO 1000
1150 RETURN
1160 REM---LIST DIRECTORY---
1170 PRINT "FILE  HR.     DAY                  STATUS"
1180 PRINT
1190 F=1
1200 IF D[F]=0 THEN 1250
1210 GOSUB 1260
1220 PRINT F;H1;D1,F$
1230 F=F+1
1240 GOTO 1200
1250 RETURN
1260 IF D[F]>24000 THEN 1340
1270 H1=INT(D[F]/1000)
1280 D1=D[F]-H1*1000
1290 F$="INCOMPLETE"
1300 IF D1<367 THEN 1330
1310 D1=D1-366
1320 F$="COMPLETE"
1330 RETURN
1340 H1=0
1350 D1=0
1360 IF D[F] >= 26000 THEN 1390
1370 F$="START UP"
1380 RETURN
1390 F$="SHUT DOWN"
1400 RETURN
1410 REM
1420 D=INT(I*S/24)+1
```

```
1430 REM---YOUR PROGRAM GOES HERE----
1440 REM---DATA IS IN W,X,Y AND Z-----
1450 REM---RESULT GOES INTO A(I,1)---
1460 REM---IF A(I,2)=1 DATA IS INVALID---
1470 REM---FOR EXAMPLE---
1480 FOR I=1 TO 256
1490 IF A[I,2] THEN 1510
1500 A[I,1]=W[I]+X[I]+Y[I]+Z[I]
1510 NEXT I
1520 REM---NOW STORE A---
1530 REM---FILE SIZE 1024 WORDS---
1540 REM---FOR EXAMPLE---
1550 REM STORE  DATA 12,A
1560 REM
1570 REM
1580 REM
1590 REM
1600 REM
1610 REM
1620 REM
1630 REM---CALCULATE APPROXIMATE YMIN AND YMAX---
1640 H1=-9999
1650 L1=9999
1660 FOR I=1 TO 256
1670 IF A[I,2] THEN 1720
1680 IF A[I,1]<H1 THEN 1700
1690 H1=A[I,1]
1700 IF A[I,1]>L1 THEN 1720
1710 L1=A[I,1]
1720 NEXT I
1730 REM---PRINT OUT SOME USEFULL INFO. ---
1740 PRINT "FIRST DAY="D3
1750 PRINT "LAST DAY="D4
1760 PRINT "NUMBER OF DAYS="D
1770 PRINT "DATA FOR EVERY"S"HOURS"
1780 PRINT "HIGHEST Y VALUE="H1
1790 PRINT "LOWEST Y VALUE="L1
1800 END
```

## 6.4 Plotter Program

NOTE: This program will not load into memory on a machine which does not contain the plotter control "ROM". Any attempt to do so will result in an "Error 1".

This program was written specifically to plot data created by the data extraction program previously discussed. This is not a general purpose plotting program, however, many desirable features have been incorporated and the source listing of this program could be used as a guide in writing a more general purpose program.

No matter what size of paper (within reason) is used, the program will place the graph on the paper so that the left-hand margin is .3 times the width of the graph and the right-hand margin is .2 of the width. The graph is placed on the paper with .2 of its height left for each margin above and below the graph. The bottom margin will, however, be a little wider since the bottom 3/4 of an inch of paper cannot be reached by the pen.

The horizontal or X-axis is divided into 1 hour units by the program. Depending on the length of the run and the size of the paper "Tics" will be placed at 1,2,4,6,12 or 24 hour intervals along the bottom X-axis. These tics will be at approximately .15 inch spacing. The top X-axis always has "Tics" at one day intervals.

The vertical or Y-axis will have "Tics" at about .1 inch spacing. The program chooses a spacing so that the "Tics" represent multiples of 1, 2 or 5 for ease of interpreting the graph. The right-hand Y-axis will have 1 "Tic" for every 10 on the left-hand margin.

In addition to these small "Tics" on the X and Y axes, larger "Tics" will be placed on the left and bottom axes. On the left axis, these are spaced every 10 "Tics" and on the bottom these major "Tics" represent whole days.

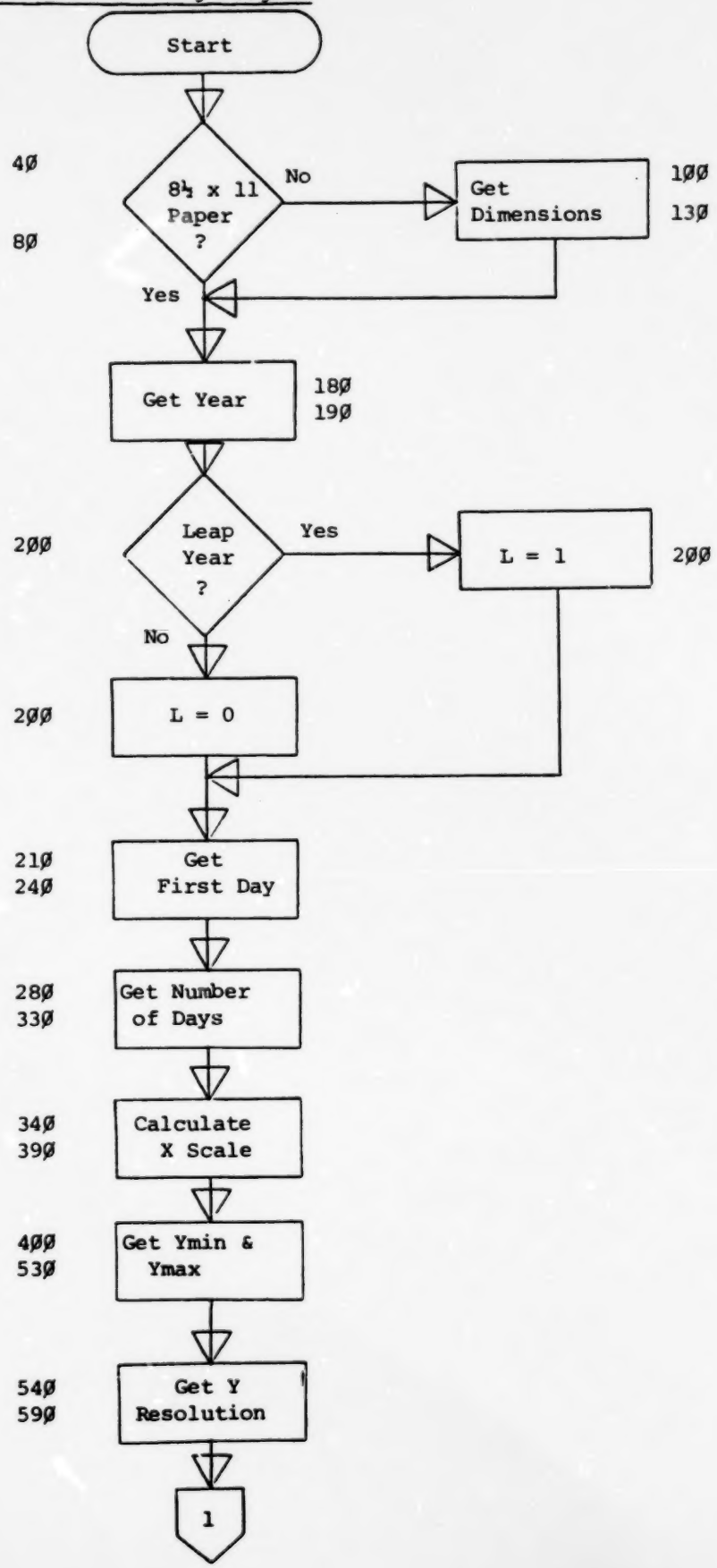
On the Y-axis, each major "Tic" is labelled with a number and on the X-axis, the days of the year are labelled at about 1 inch intervals.

The limits of the Y-axis should be chosen using the print out from the data extraction program as a guide. These limits should be between -9999 and +9999 and the resolution should be chosen depending on the nature of the data. Only one decimal place is provided so the user routine in the data extraction program should give a result which will fit these restrictions. For example, feed rate should be converted to GM/HR rather than KG/HR. The Y-axis label can contain a multiple such as "Feed rate KG/HR x 1000".

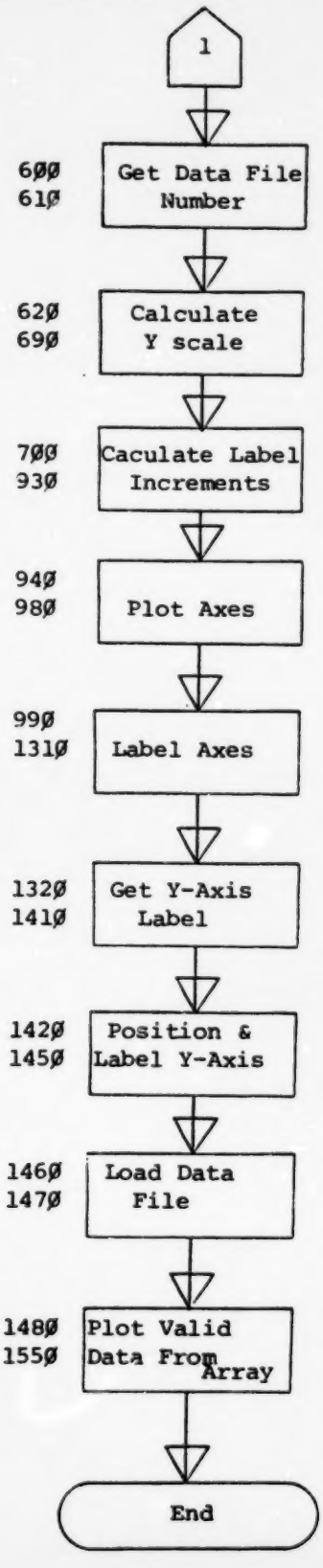
The X-axis is always labelled "Days of the year" but the Y-axis can be given any label. This label will be automatically centered on the Y-axis and placed just to the left of the longest numerical label. A run number or title may also be typed in and the program will place it just above the centre of the graph.

The program will then load the requested data file and plot it, leaving spaces wherever the data is invalid.

File 2 - Plotting Program



File 2 - Plotting Program (cont'd)



File 2

```
10 DIM A$[80],AS[256,2]
20 DEG
30 REM---GET PAPER SIZE---
40 DISP "8.5 X 11 PAPER";
50 INPUT A$
60 IF A$[1,1]="N" THEN 100
70 H=8.5
80 W=11
90 GOTO 140
100 DISP "HEIGHT OF PAPER (INCHES)";
110 INPUT H
120 DISP "WIDTH OF PAPER (INCHES)";
130 INPUT W
140 R1=H/W
150 REM---ASPECT RATIO---
160 R2=1.7
170 REM---LEAP YEAR ?---
180 DISP "ENTER YEAR OF FIRST DAY";
190 INPUT Y
200 L=((Y/4=INT(Y/4)) AND  NOT (Y/100=INT(Y/100))) OR (Y/400=INT(Y/400))
210 REM---GET X AXIS INFO.---
220 DISP "ENTER FIRST DAY";
230 INPUT D1
240 IF D1>0 AND D1<365+L THEN 280
250 DISP "NOT A VALID DAY"
260 GOSUB 1570
270 GOTO 220
280 DISP "ENTER NUMBER OF DAYS";
290 INPUT D
300 IF D>0 AND D<23 THEN 380
310 DISP "DAYS MUST BE >0 AND <23"
320 GOSUB 1570
330 GOTO 280
340 REM---TOTAL PAPER WIDTH=1.5---
350 REM---LEFT MARGIN=.3---
360 REM---RIGHT MARGIN=.2---
370 REM---LEAVING 1 FOR GRAPH---
380 X1=(D1-1)*24-0.3*D*24
390 X2=(D1-1)*24+24*D*1.2
```

```
400 REM---GET Y AXIS INFO. ---
410 REM---USE NICE ROUND NUMBERS---
420 DISP "ENTER LOW Y";
430 INPUT L1
440 IF L1>-10000 AND L1<10000 THEN 480
450 DISP "LOW Y OUT OF BOUNDS"
460 GOSUB 1570
470 GOTO 420
480 DISP "ENTER HIGH Y";
490 INPUT H1
500 IF H1>-10000 AND H1<10000 THEN 540
510 DISP "HIGH Y OUT OF BOUNDS"
520 GOSUB 1570
530 GOTO 480
540 DISP "ENTER RESOLUTION (.1 OR 1 OR 10)";
550 INPUT R3
560 IF R3=0.1 OR R3=1 OR R3=10 THEN 600
570 DISP "NOT A VALID RESOLUTION"
580 GOSUB 1570
590 GOTO 540
600 DISP "ENTER DATA FILE NUMBER";
610 INPUT F
620 REM---TOTAL PAPER HEIGHT=1.4---
630 REM---1 FOR THE GRAPH, .2 TOP AND BOTTOM---
640 REM---IF PAPER IS PLACED AGAINST BOTTOM EDGE---
650 REM---BOTTOM MARGIN WILL BE WIDER---
660 REM---THIS IS NORMAL---
670 Y1=L1/R3-0.2*(H1-L1)/R3
680 Y2=L1/R3+1.2*(H1-L1)/R3
690 SCALE X1,X2,Y1,Y2
700 REM---CALCULATE UNITS PER INCH---
710 R4=D*24*1.5/W
720 R5=(H1-L1)/R3*1.4/H
730 REM---CALCULATE X LABEL INC.  IN 1 HR.  UNITS---
740 S=1
750 IF S*24/R4>1 THEN 790
760 S=INT(S+1.6*S)
770 GOTO 750
780 REM---CALCULATE X TIC (1,2,4,6,12 OR 24 HOURS)---
790 T=1
800 IF T/R4>0.15 THEN 840
810 T=2*T-2*(T=4)
820 GOTO 800
830 REM---CALCULATE Y TIC ALWAYS MULTIPLE OF 1,2 OR 5---
840 P=-1
850 T1=1
860 IF T1*10^P/R5>0.1 THEN 910
870 T1=INT(T1+1.6*T1)
880 IF T1 <= 5 THEN 860
890 P=P+1
900 GOTO 850
910 T1=T1*10^P
920 REM---LABEL Y AXIS EVERY 10 TICS---
930 L2=T1*10
```

```
940 REM---DRAW THE AXES---
950 XAXIS L1/R3,T,(D1-1)*24,(D1-1+D)*24
960 YAXIS (D1-1+D)*24,L2,L1/R3,H1/R3
970 XAXIS H1/R3,-24,(D1-1+D)*24,(D1-1)*24
980 YAXIS (D1-1)*24,-T1,H1/R3,L1/R3
990 REM---CALCULATE SIZE OF MAJOR TICS---
1000 E=(H1-L1)/R3*1.4/H*0.2
1010 D2=D*24*1.5/W*0.2
1020 REM---SET SIZE OF CHARACTERS---
1030 LABEL (*,20/H,R2,0,R1)
1040 REM---MAJOR TIC EVERY DAY---
1050 FOR I=D1 TO D1-1+D
1060 PLOT I*24,L1/R3,-2
1070 IPLOT 0,E
1080 IPLOT 0,-E
1090 NEXT I
1100 REM---NUMBER X AXIS---
1110 FOR I=D1 TO D1-1+D STEP S
1120 PLOT I*24,L1/R3,-2
1130 CPLOT -3.3+INT(LGT(I-(365+L)*(I>365+L)))*0.5,-1.3
1140 LABEL (1150)I-(365+L)*(I>365+L)
1150 FORMAT F4.0
1160 PEN
1170 NEXT I
1180 REM---NUMBER Y AXIS---
1190 FOR I=L1 TO H1 STEP L2*R3
1200 PLOT (D1-1)*24,I/R3,-2
1210 IPLOT D2,0
1220 IPLOT -D2,0,1
1230 CPLOT -7.5,-0.3
1240 LABEL (1250)I
1250 FORMAT F7.1
1260 PEN
1270 NEXT I
1280 REM---LABEL X AXIS "DAY OF THE YEAR---
1290 PLOT (D1-1+D*0.5)*24,L1/R3
1300 CPLOT -7.5,-2.3
1310 LABEL (*)"DAY OF THE YEAR"
1320 REM---TYPE RUN NO. OR TITLE---
1330 DISP "ENTER RUN NUMBER";
1340 INPUT A$
1350 PLOT (D1-1+D*0.5)*24,H1/R3
1360 CPLOT -LEN(A$)*0.5,1
1370 LABEL (*)A$
1380 REM---LABEL CENTERED ON Y AXIS---
1390 REM---AND JUST TO LEFT OF HIGEST NUMBER---
1400 DISP "ENTER Y-LABEL";
1410 INPUT A$
1420 LABEL (*,20/H,R2,90,R1)
1430 PLOT (D1-1)*24,L1/R3+(H1-L1)/R3*0.5,1
1440 CPLOT -LEN(A$)*0.5,(4.3+(L1<0)+INTLGT(H1))/R2
1450 LABEL (*)A$
```

```
1460 REM---GET THE PRE-FORMATTED DATA---
1470 LOAD  DATA F,A
1480 FOR I=1 TO 256
1490 REM---PLOT ONLY VALID DATA---
1500 IF A[I,2] OR I>(D*24)/INT(D*24/256)+1 THEN 1540
1510 PLOT I+(D1-1)*24,A[I,1]/R3
1520 NEXT I
1530 END
1540 PEN
1550 GOTO 1520
1560 REM---DISPLAY SUBROUTINE---
1570 WAIT 3000
1580 RETURN

M
```